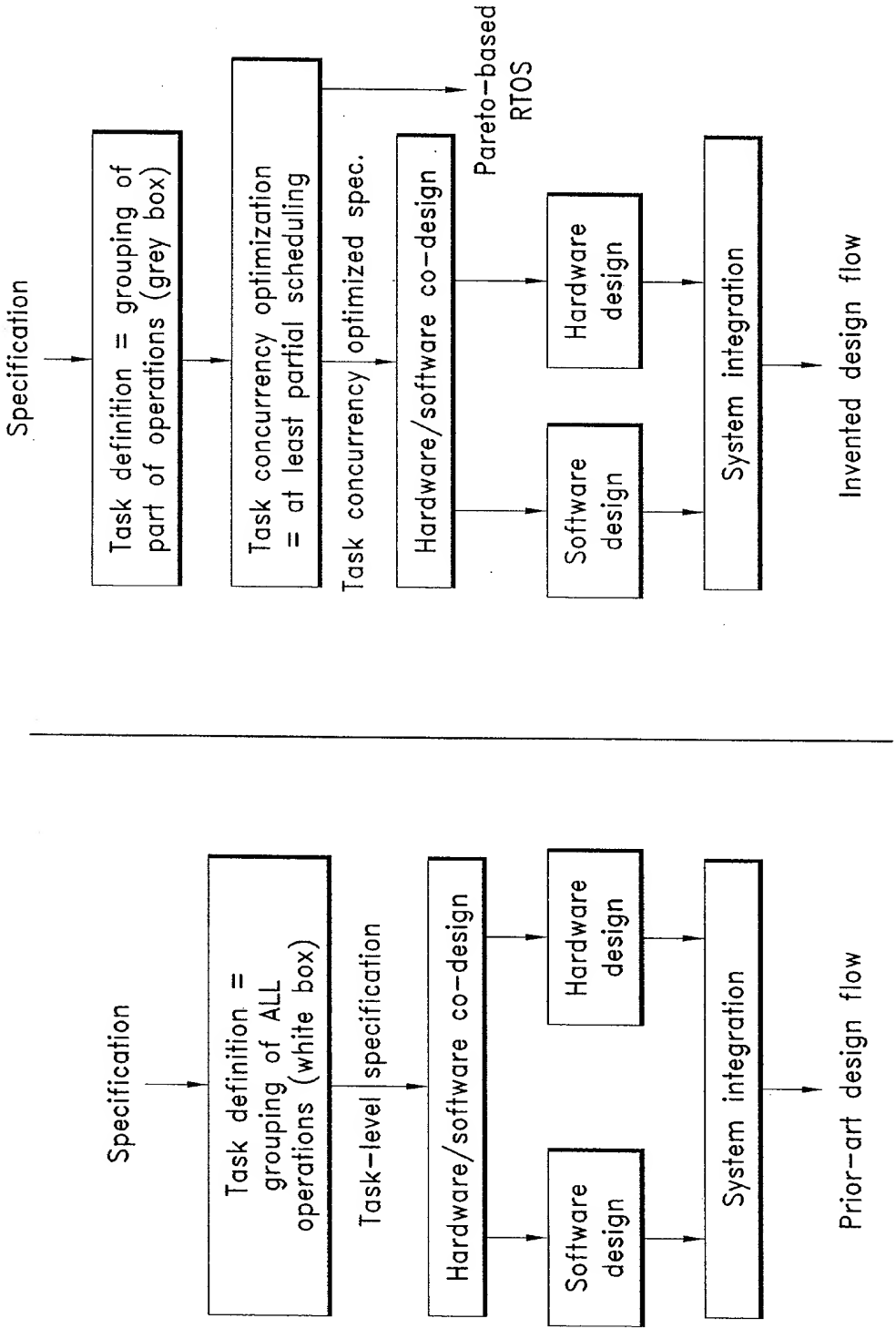


FIG. 1
(PRIOR ART)

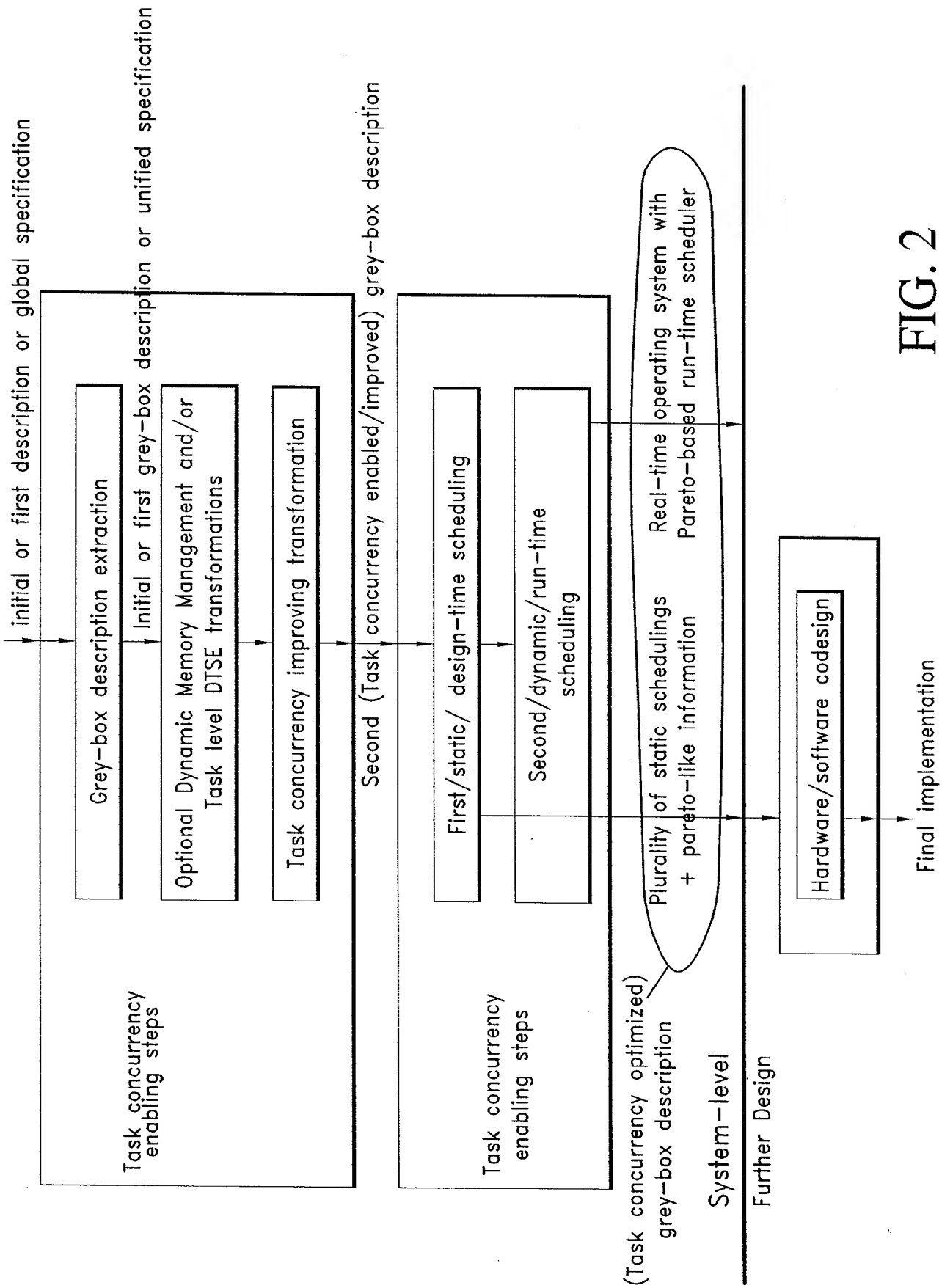

initial or first description or global specification
Grey-box description extraction
Initial or first grey-box description or unified specification
Optional Dynamic Memory Management and/or Task level DTSE transformations
Task concurrency improving transformation
Task concurrency enabling steps
Second (Task concurrency enabled/improved) grey-box description
First/static/ design-time scheduling
Second/dynamic/run-time scheduling
Task concurrency enabling steps
Plurality of static schedulings + pareto-like information
Real-time operating system with Pareto-based run-time scheduler
(Task concurrency optimized) grey-box description
System-level
Further Design
Hardware/software codesign
Final implementation

FIG. 2

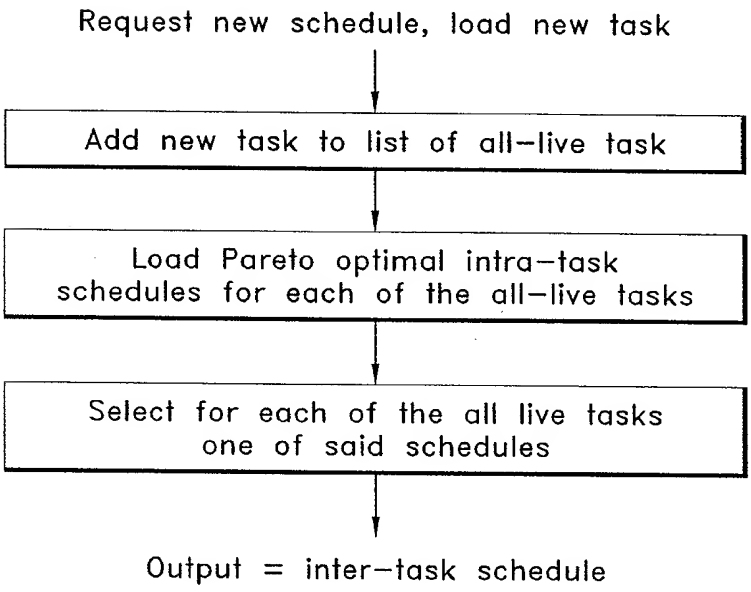

Request new schedule, load new task
Add new task to list of all-live task
Load Pareto optimal intra-task schedules for each of the all-live tasks
Select for each of the all live tasks one of said schedules
Output = inter-task schedule

FIG. 3

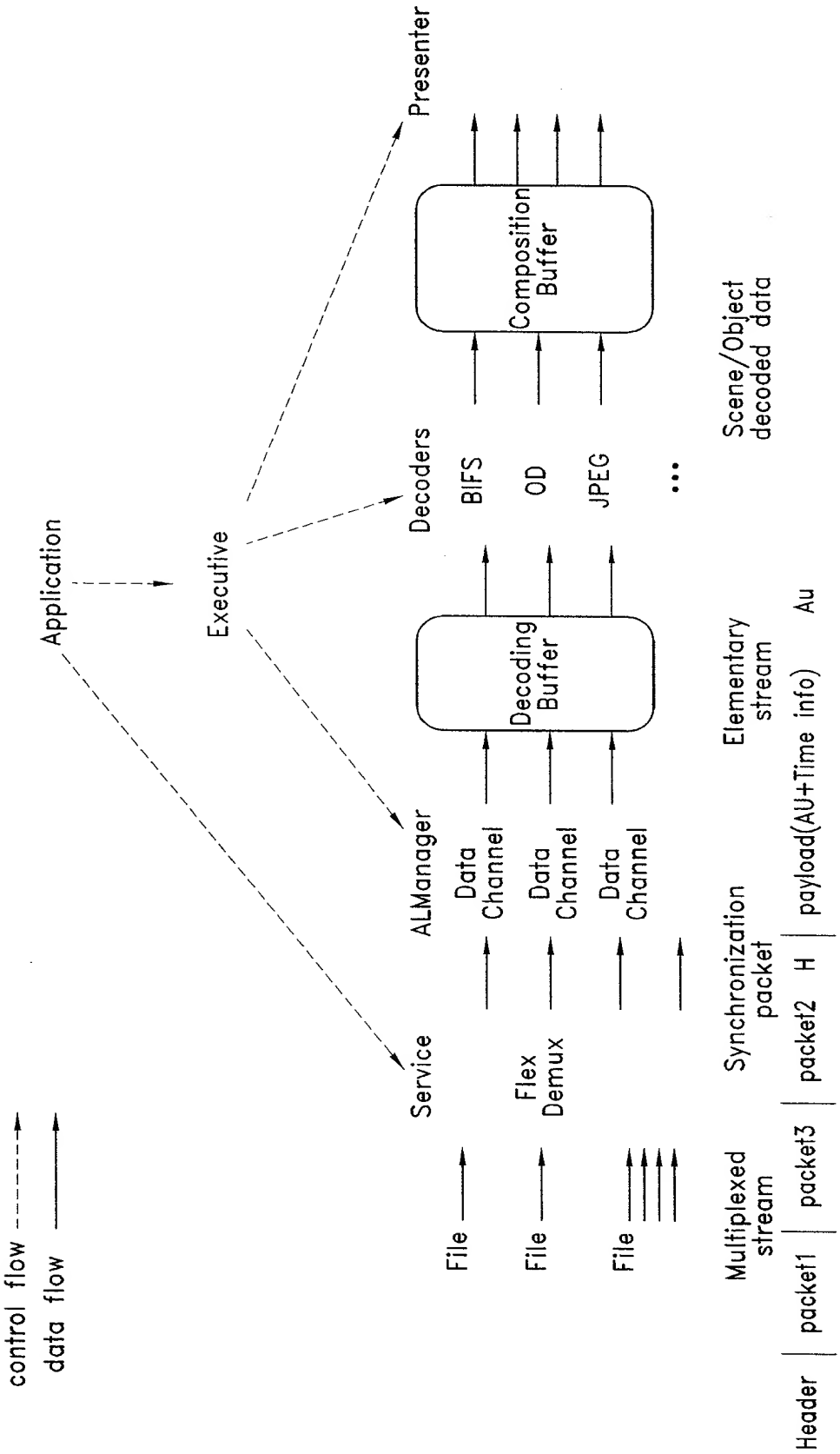

control flow
data flow
Application
Executive
Presenter
Decoders
BIFS
OD
JPEG
...
Composition Buffer
Scene/Object decoded data
Decoding Buffer
Elementary stream
Au
ALManager
Data Channel
Data Channel
Data Channel
Service
Flex Demux
Synchronization packet
H
payload(AU+Time info)
File
File
File
Multiplexed stream
Header
packet1
packet3
packet2

FIG. 4

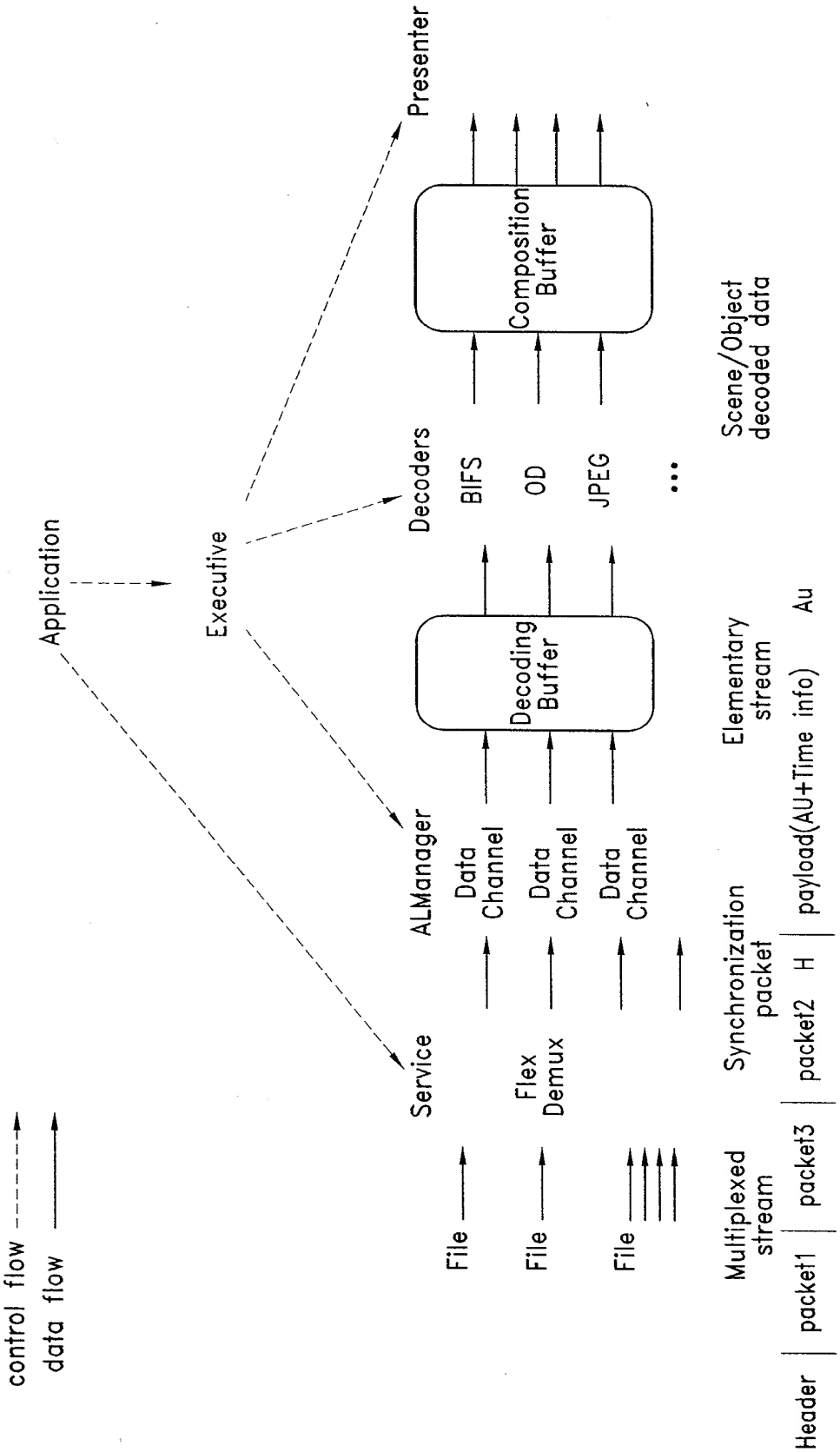

control flow
data flow
Application
Executive
Presenter
Decoders
BIFS
OD
JPEG
...
Composition Buffer
Scene/Object decoded data
Decoding Buffer
Elementary stream
ALManager
Data Channel
Data Channel
Data Channel
Service
Flex Demux
Synchronization packet
File
File
File
Multiplexed stream
Header
packet1
packet3
packet2
H
payload(AU+Time info)
Au

FIG. 5

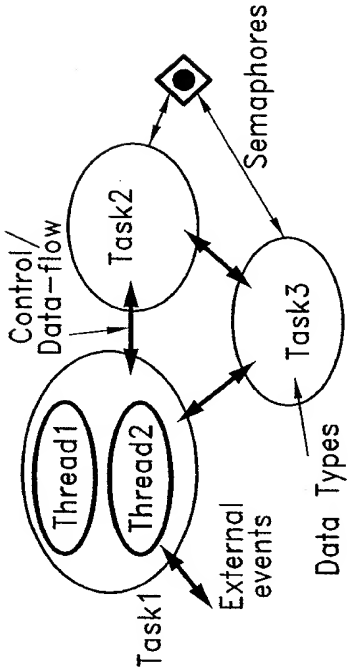

Semaphores
Control/
Data-flow
Task2
Task3
Thread1
Thread2
External
events
Data Types
Task1

FIG. 6

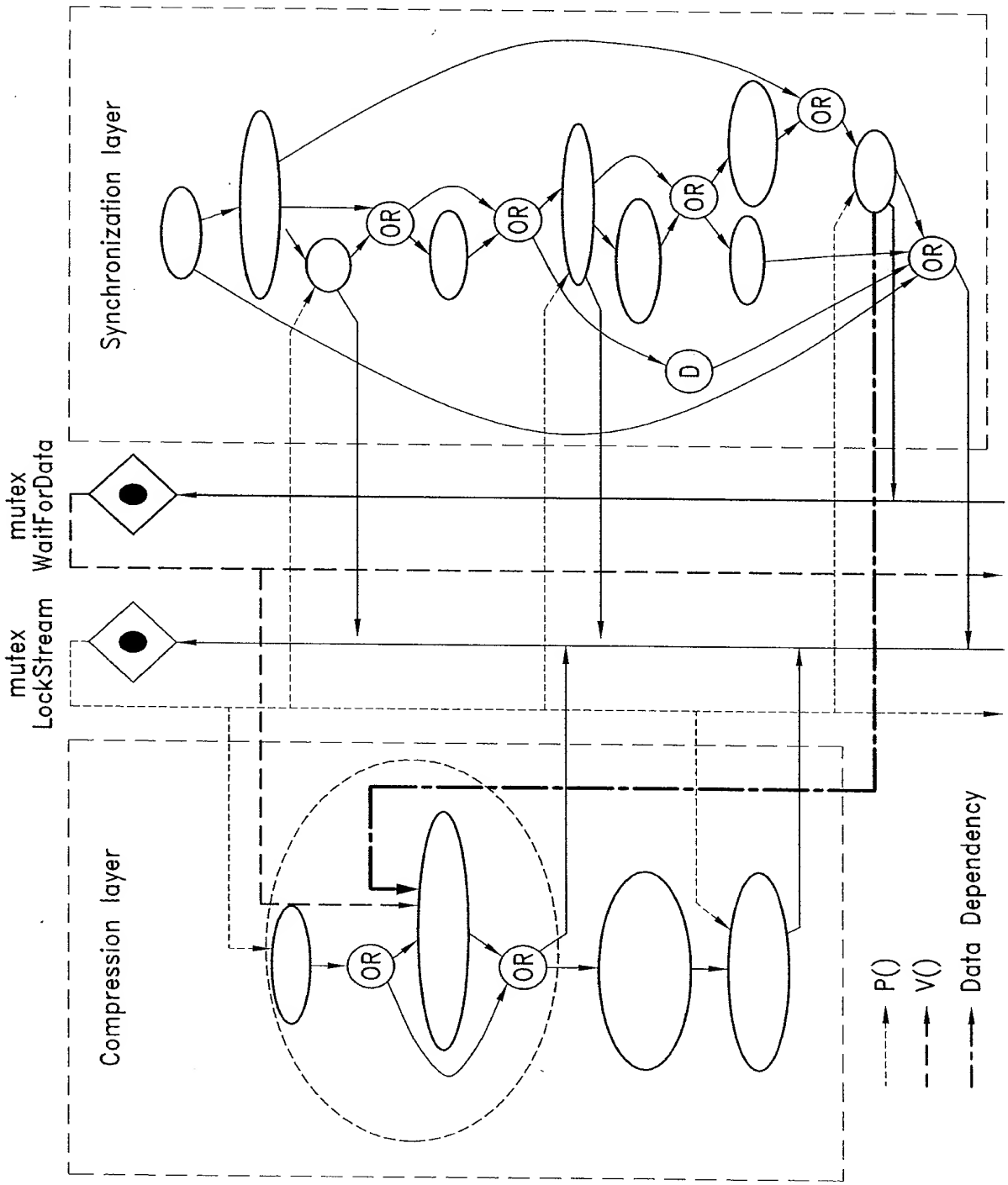

Synchronization layer
OR
D
mutex WaitForData
mutex LockStream
Compression layer
P()
V()
Data Dependency

FIG. 7

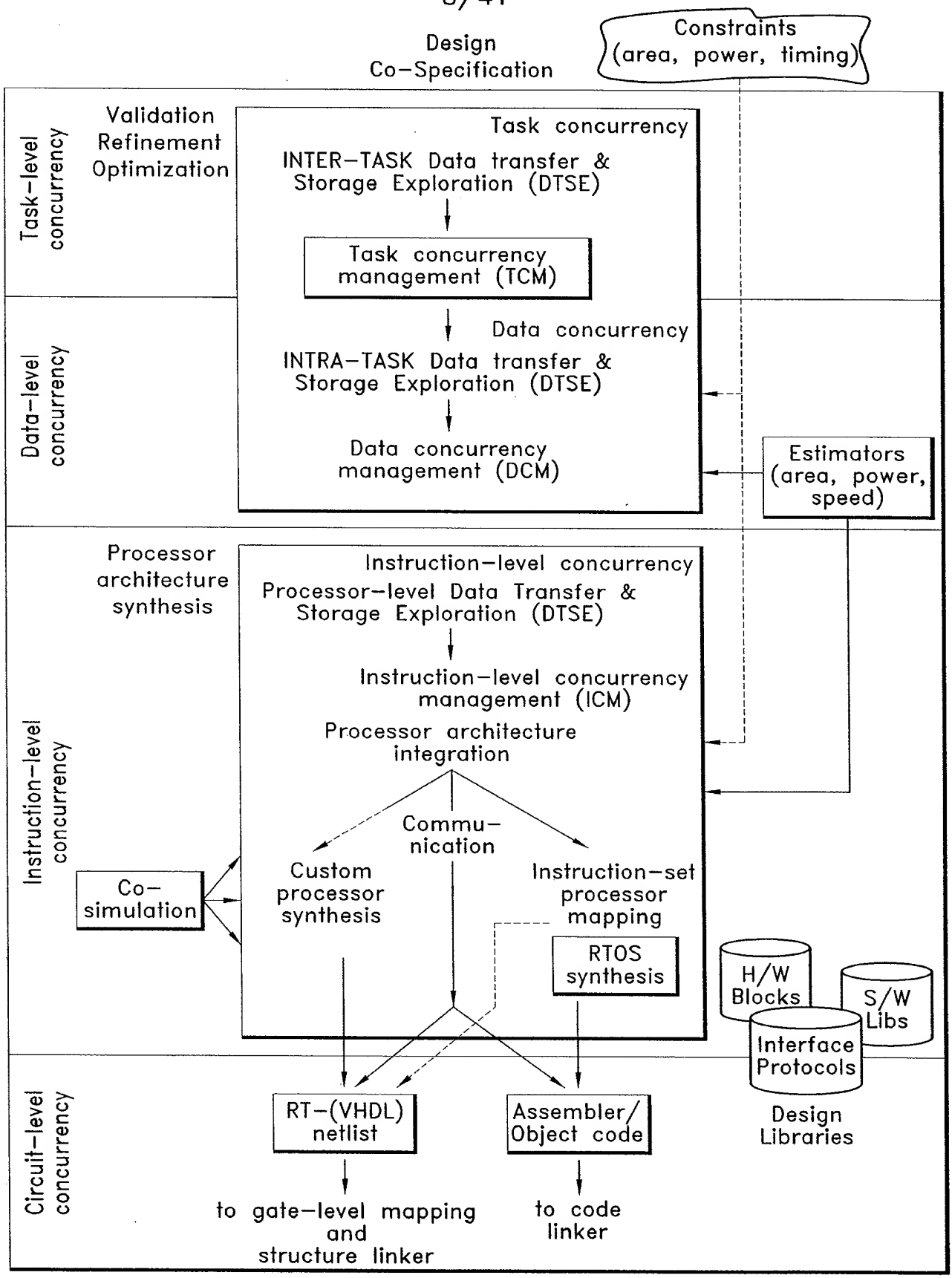

Design
Co-Specification
Constraints
(area, power, timing)
Task-level concurrency
Validation
Refinement
Optimization
Task concurrency
INTER-TASK Data transfer & Storage Exploration (DTSE)
Task concurrency management (TCM)
Data concurrency
Data-level concurrency
INTRA-TASK Data transfer & Storage Exploration (DTSE)
Data concurrency management (DCM)
Estimators (area, power, speed)
Processor architecture synthesis
Instruction-level concurrency
Processor-level Data Transfer & Storage Exploration (DTSE)
Instruction-level concurrency management (ICM)
Processor architecture integration
Instruction-level concurrency
Commu-nication
Co-simulation
Custom processor synthesis
Instruction-set processor mapping
RTOS synthesis
H/W Blocks
S/W Libs
Interface Protocols
Design Libraries
Circuit-level concurrency
RT-(VHDL) netlist
Assembler/ Object code
to gate-level mapping and structure linker
to code linker

FIG. 8

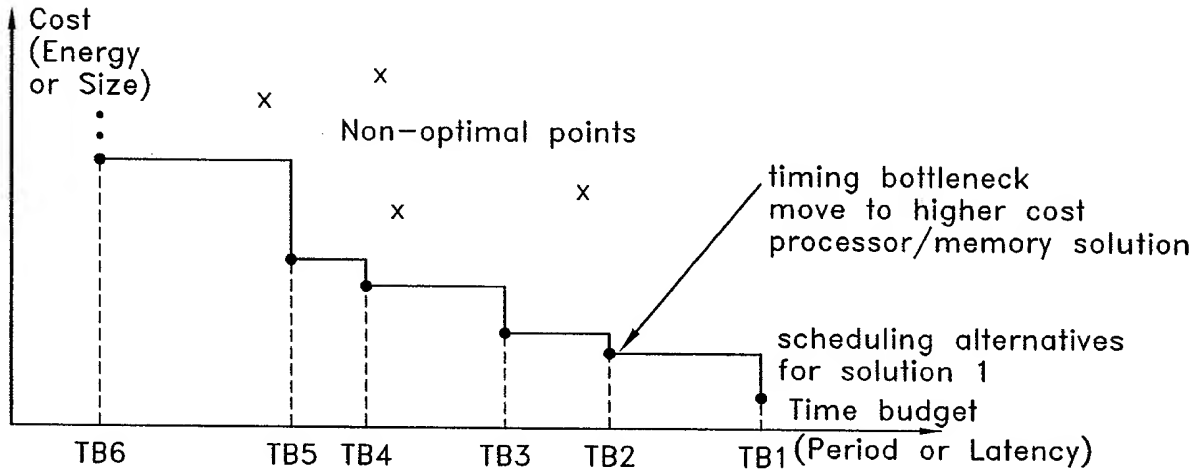


FIG. 9

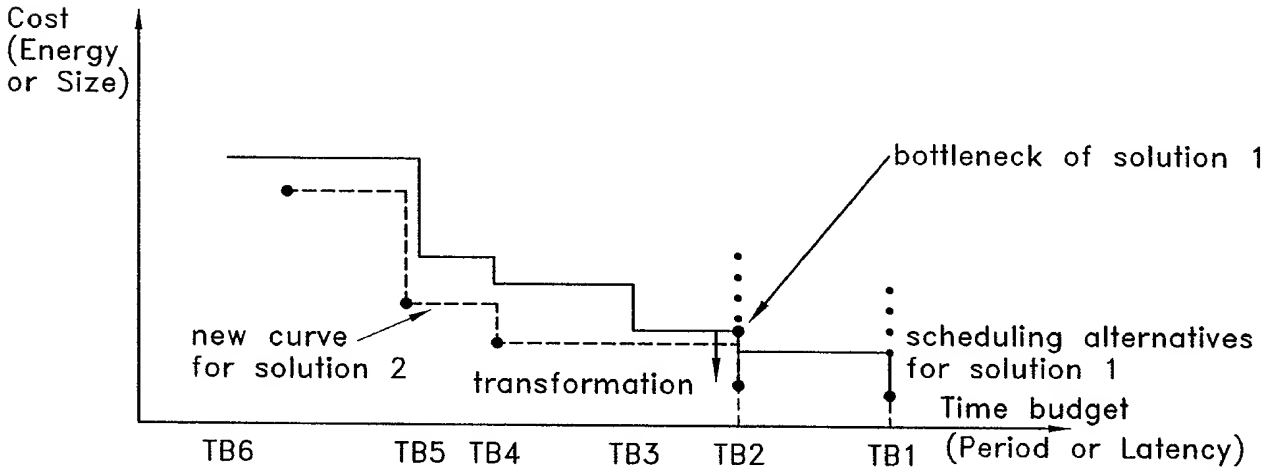


FIG. 10

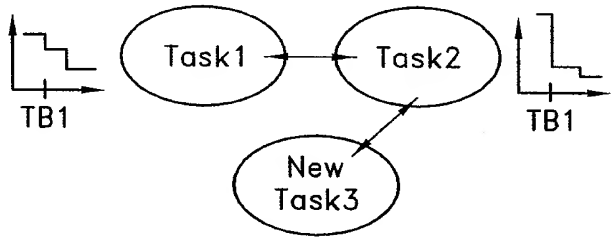


FIG. 11

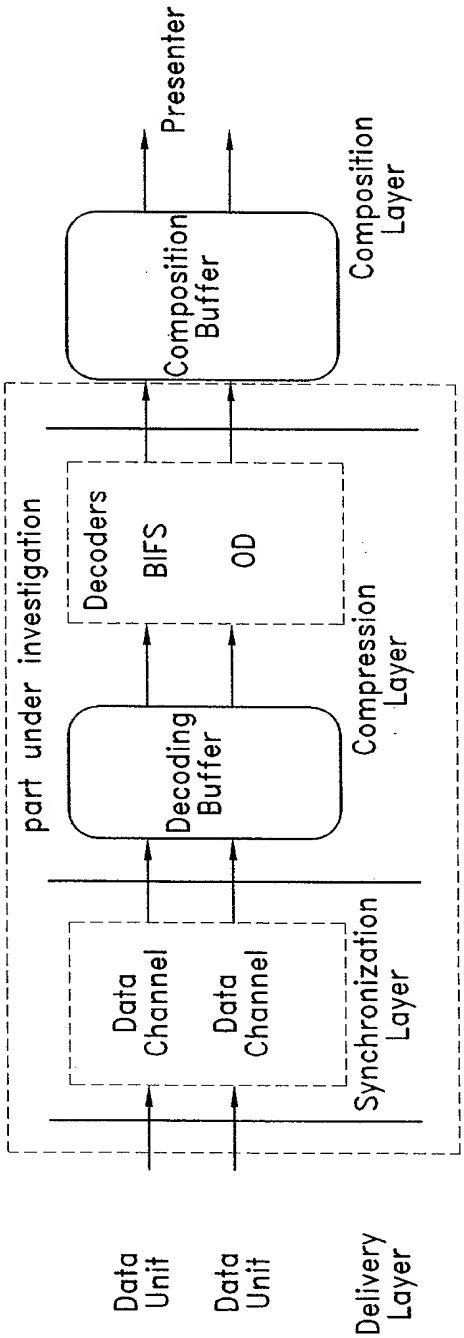

Presenter
Composition Buffer
Composition Layer
part under investigation
Decoders
BIFS
OD
Compression Layer
Decoding Buffer
Data Channel
Data Channel
Synchronization Layer
Data Unit
Data Unit
Delivery Layer

FIG. 12

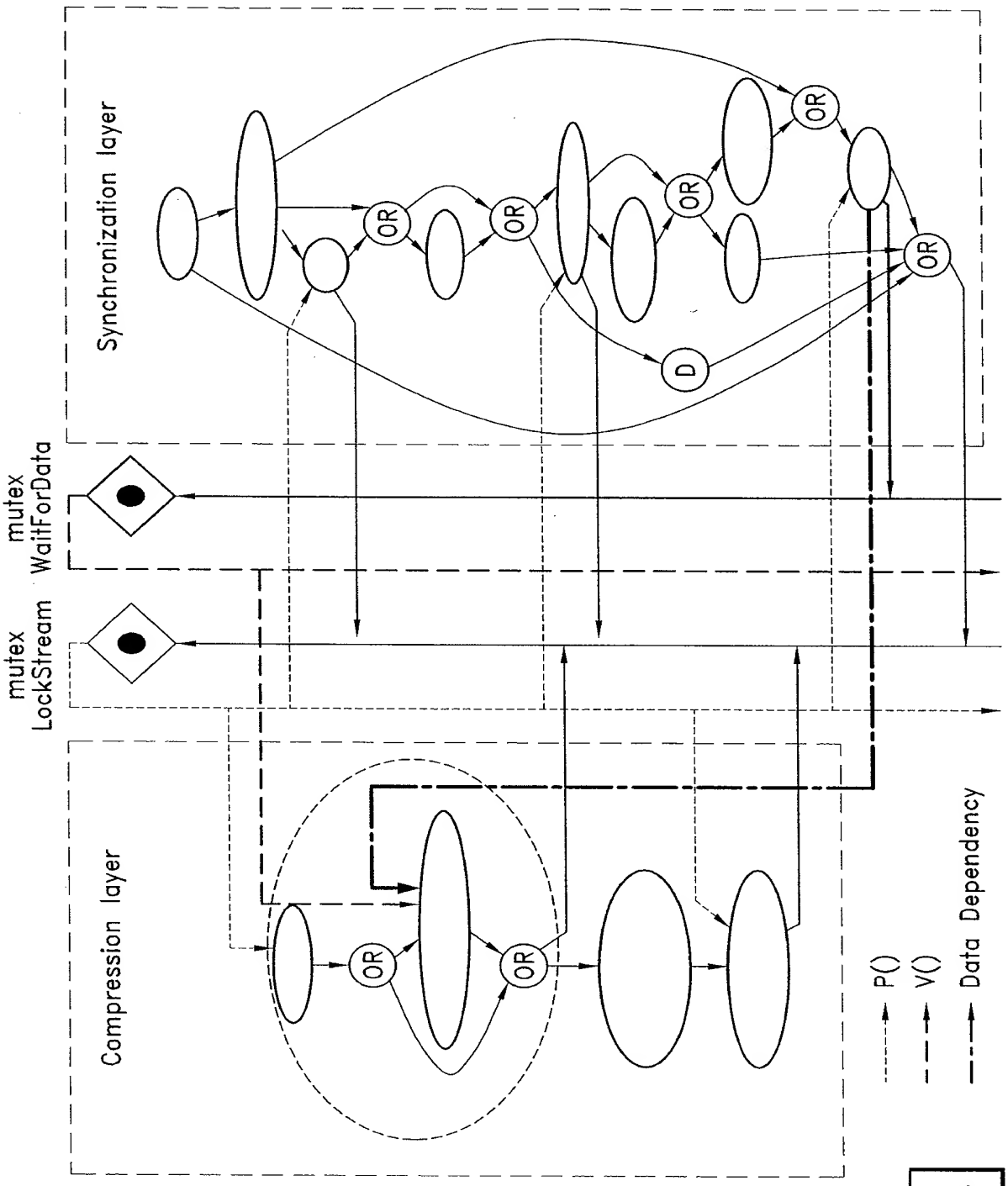


FIG. 13$_A$

FIG. 13

FIG. 13$_A$	FIG. 13$_B$

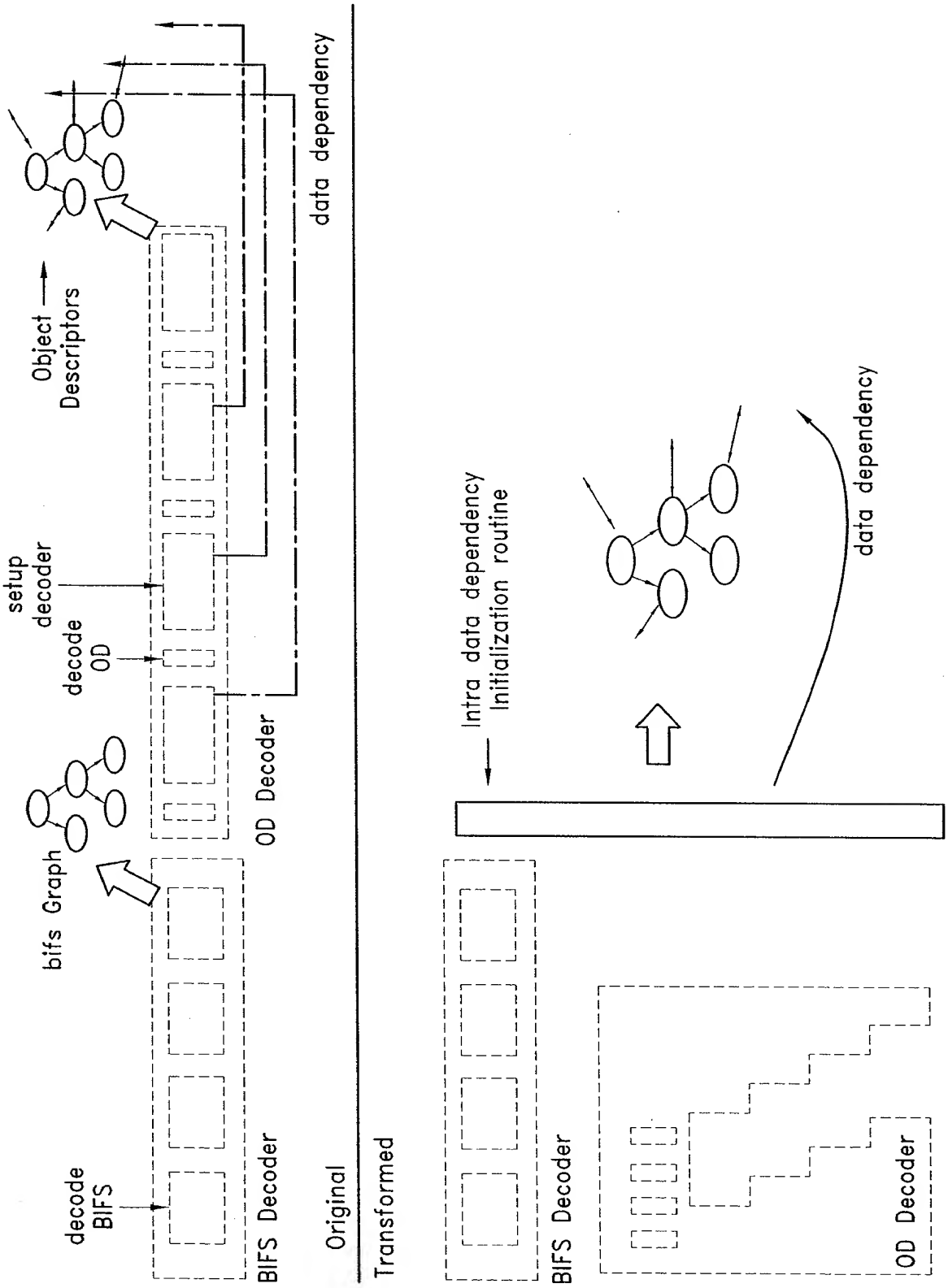

data dependency
Object Descriptors
setup decoder
decode OD
OD Decoder
bifs Graph
decode BIFS
BIFS Decoder
Original
Transformed
Intra data dependency Initialization routine
data dependency
BIFS Decoder
OD Decoder

FIG. 13_B

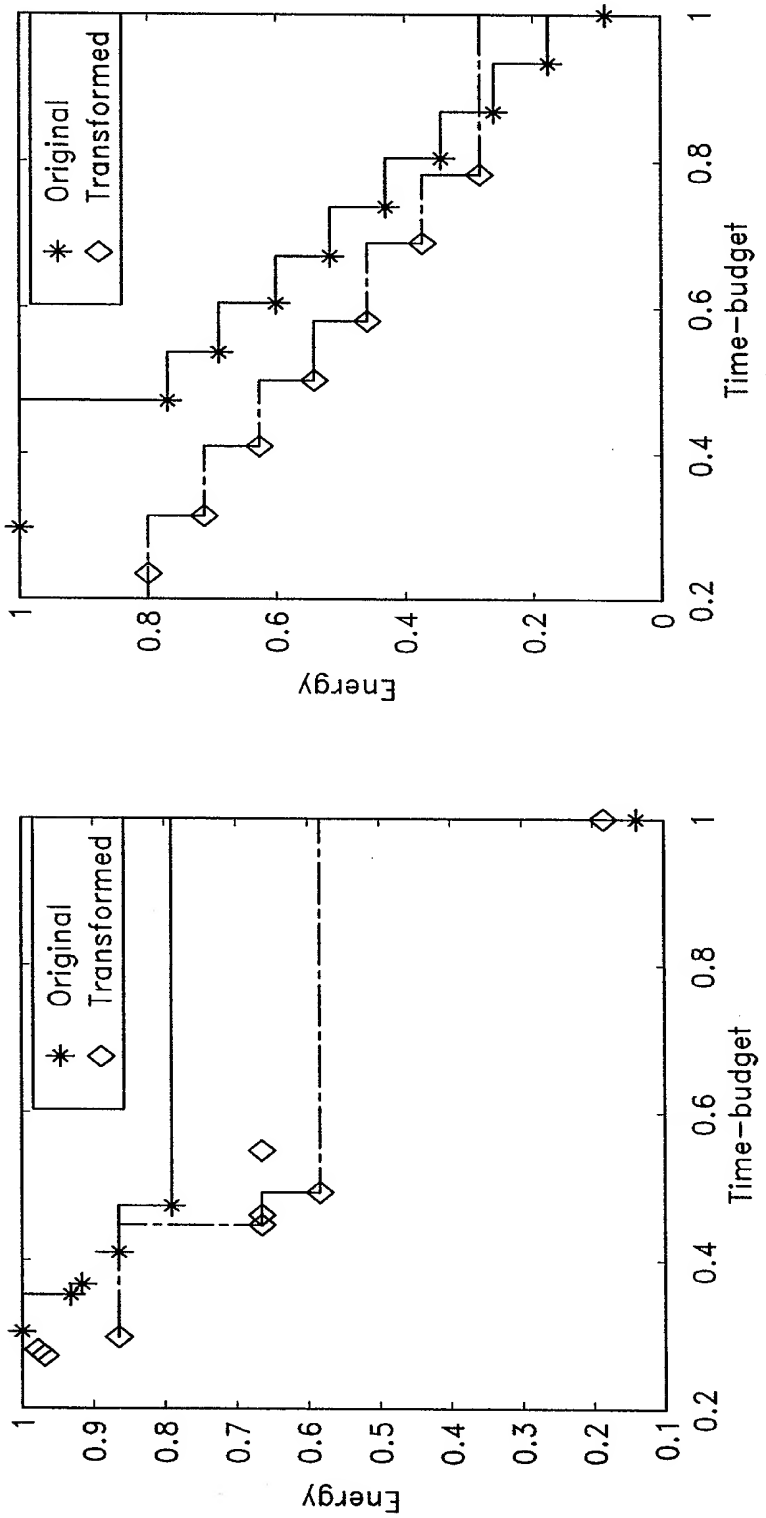

Original
Transformed
Energy
Time-budget
1
0.8
0.6
0.4
0.2
0
0.2
0.4
0.6
0.8
1
Original
Transformed
Energy
Time-budget
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.2
0.4
0.6
0.8
1

FIG. 14

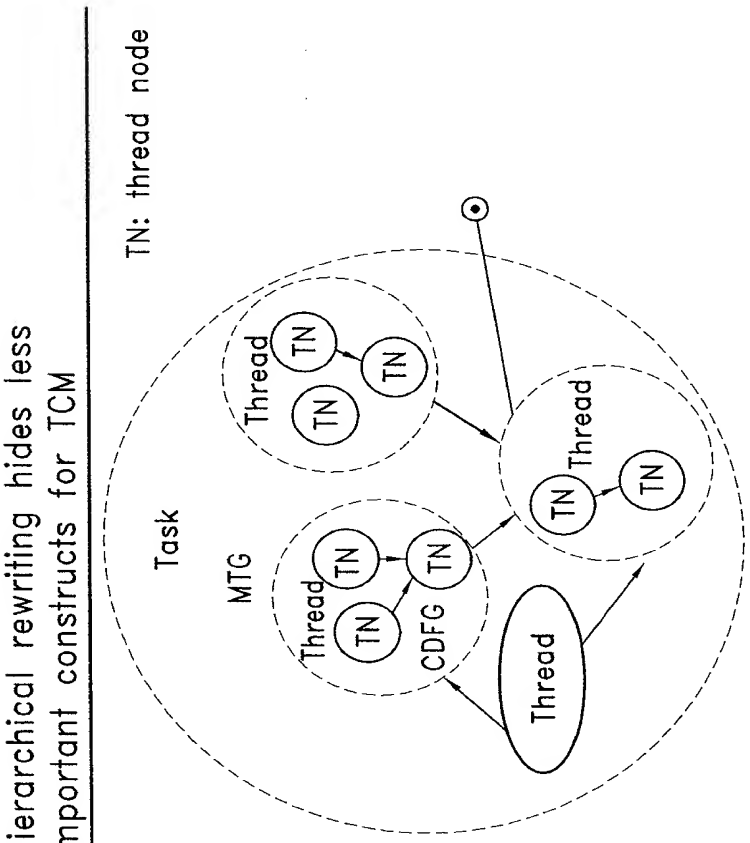

Hierarchical rewriting hides less important constructs for TCM
TN: thread node
Task
MTG
Thread
TN
CDFG
Thread
Thread
TN

FIG. 15

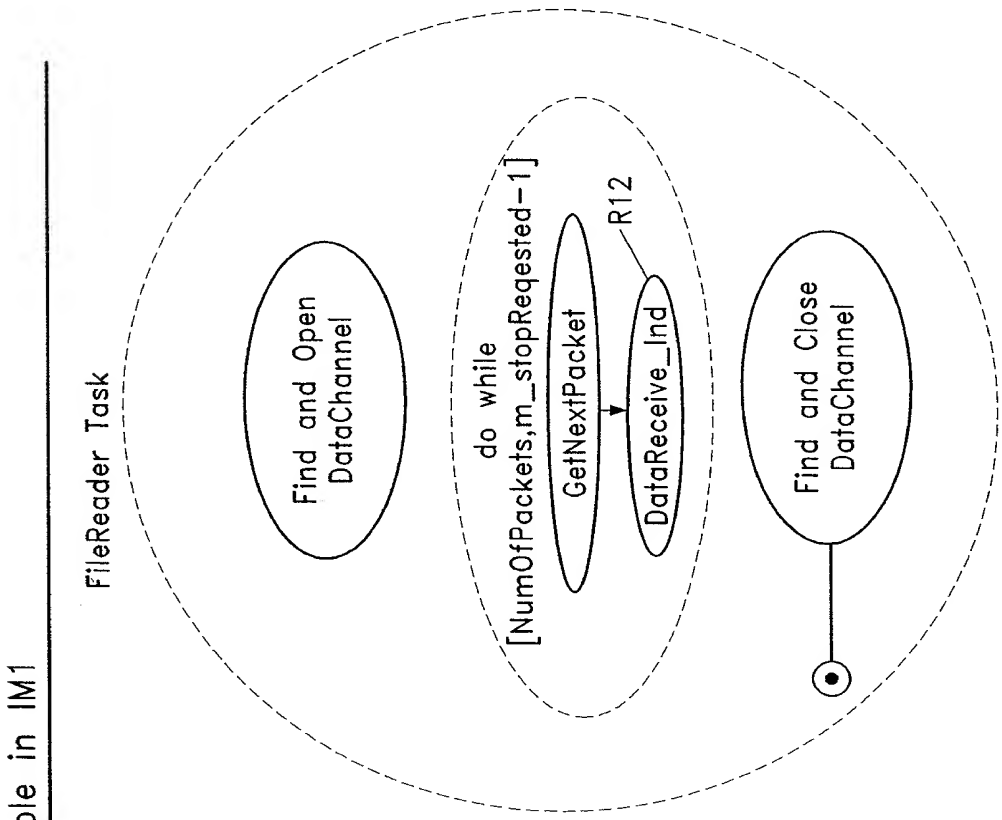

Example in IM1
FileReader Task
Find and Open DataChannel
do while [NumOfPackets,m_stopReqested-1]
GetNextPacket
R12
DataReceive_Ind
Find and Close DataChannel

FIG. 16

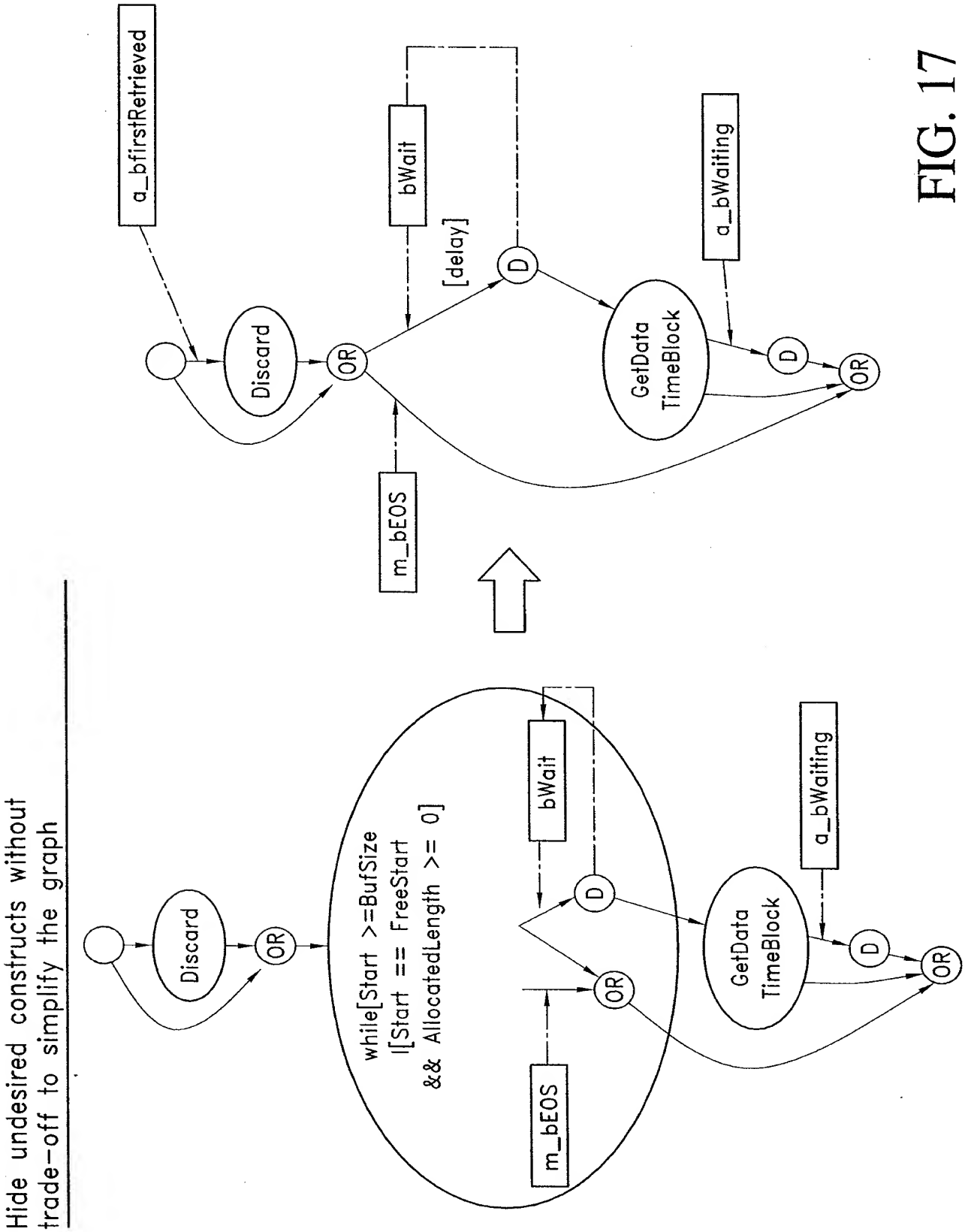

Hide undesired constructs without trade-off to simplify the graph
Discard
OR
while[Start >=BufSize
I[Start == FreeStart
&& AllocatedLength >= 0]
bWait
D
m_bEOS
OR
GetData TimeBlock
a_bWaiting
D
OR
a_bfirstRetrieved
Discard
OR
bWait
[delay]
D
GetData TimeBlock
a_bWaiting
D
OR
m_bEOS

FIG. 17

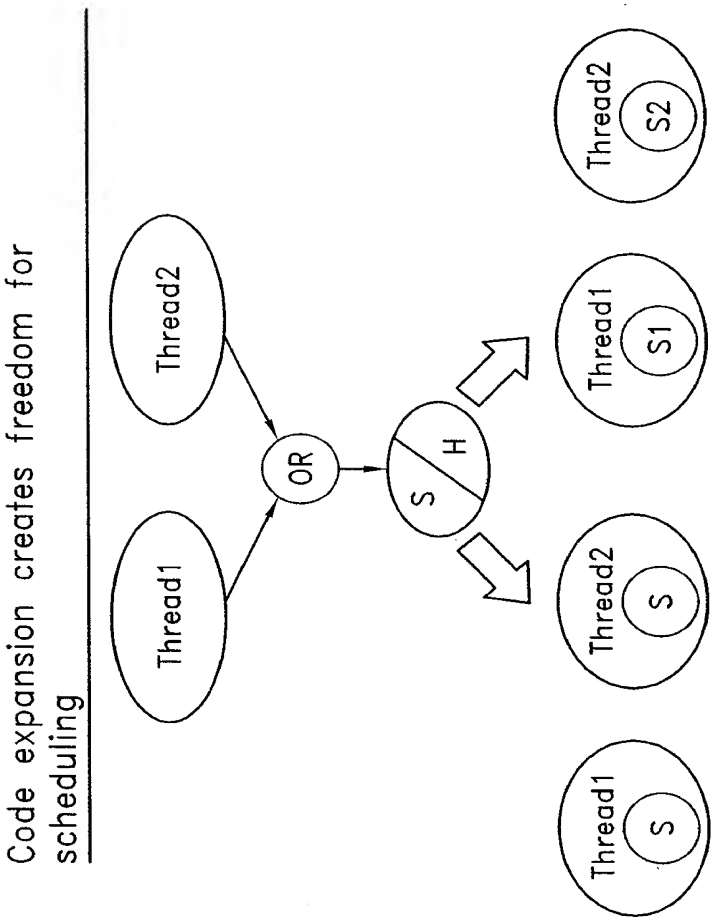

Code expansion creates freedom for scheduling
Thread2
Thread1
OR
S
H
Thread1
S1
Thread2
S2
Thread2
S
Thread1
S

FIG. 18

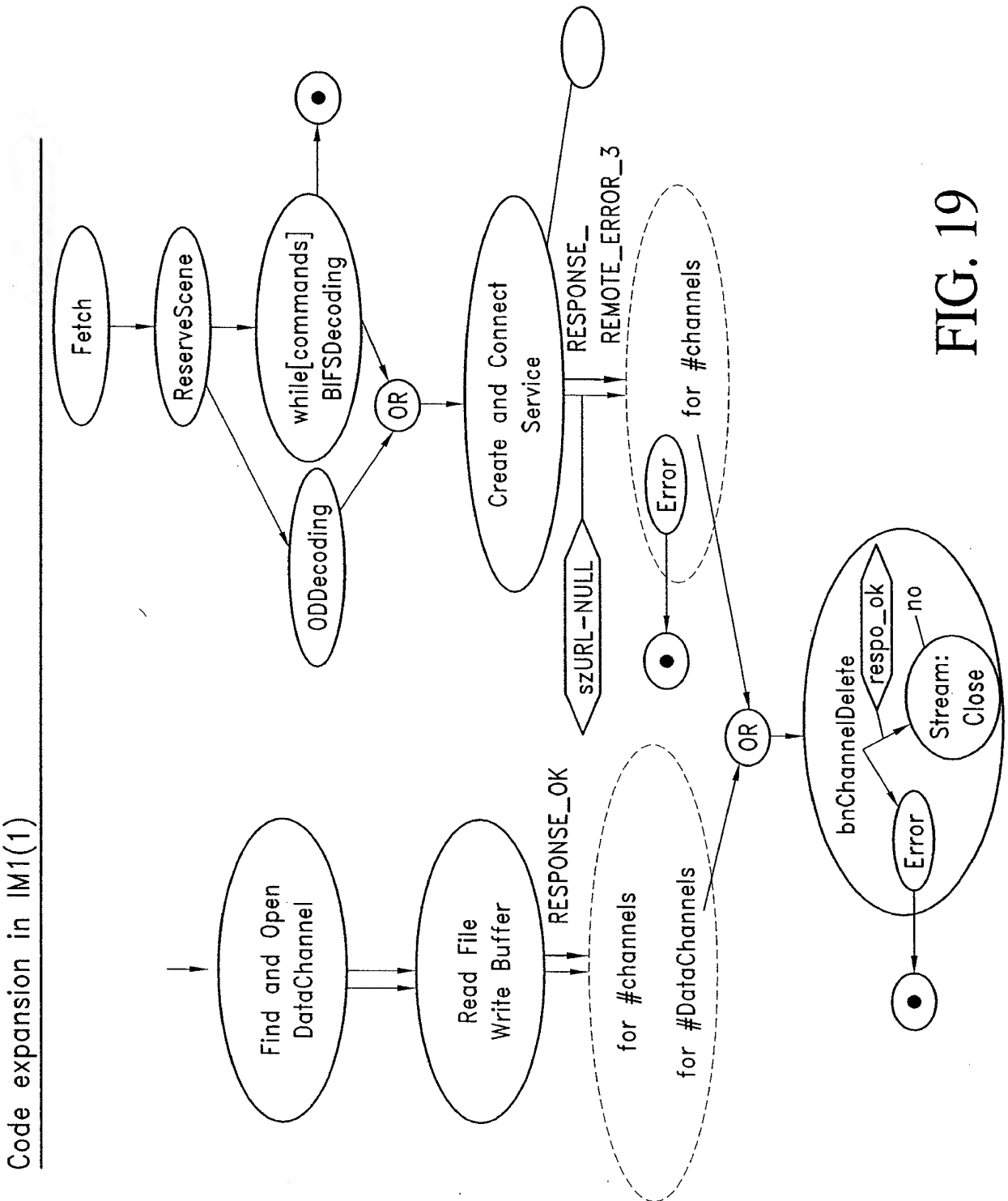
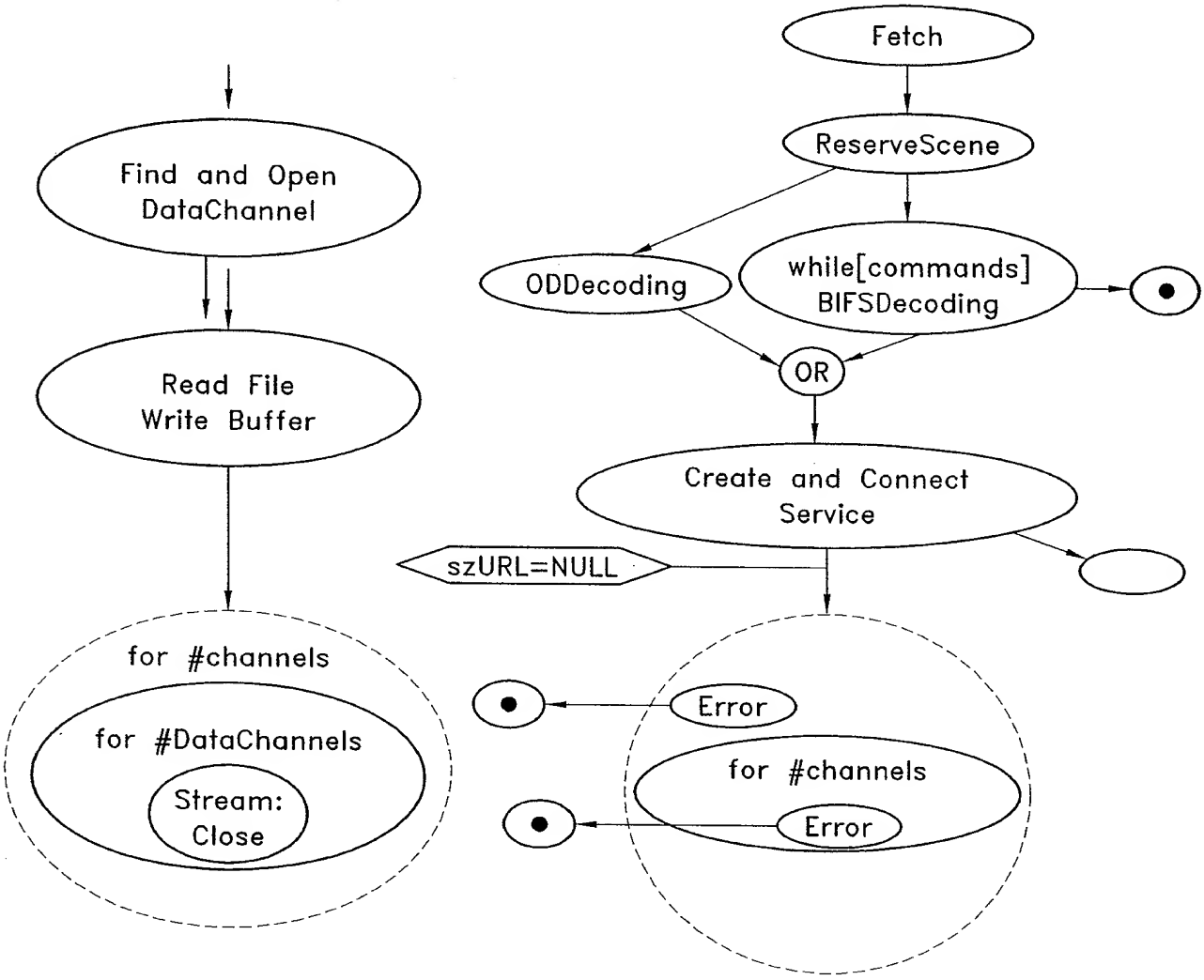

Code expansion in IM1(1)
Fetch
ReserveScene
while[commands]
BIFSDecoding
ODDecoding
OR
Create and Connect
Service
RESPONSE_
REMOTE_ERROR_3
szURL-NULL
Error
for #channels
Find and Open
DataChannel
Read File
Write Buffer
RESPONSE_OK
for #channels
for #DataChannels
OR
bnChannelDelete
respo_ok
no
Stream:
Close
Error

FIG. 19

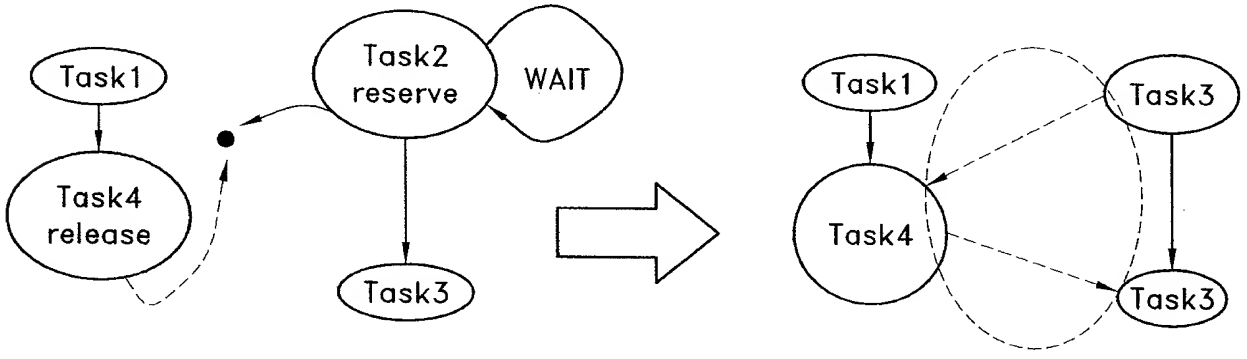
Code expansion in IM1(2)

FIG. 20



Remove constructs that make concurrency analysis difficult

FIG. 21



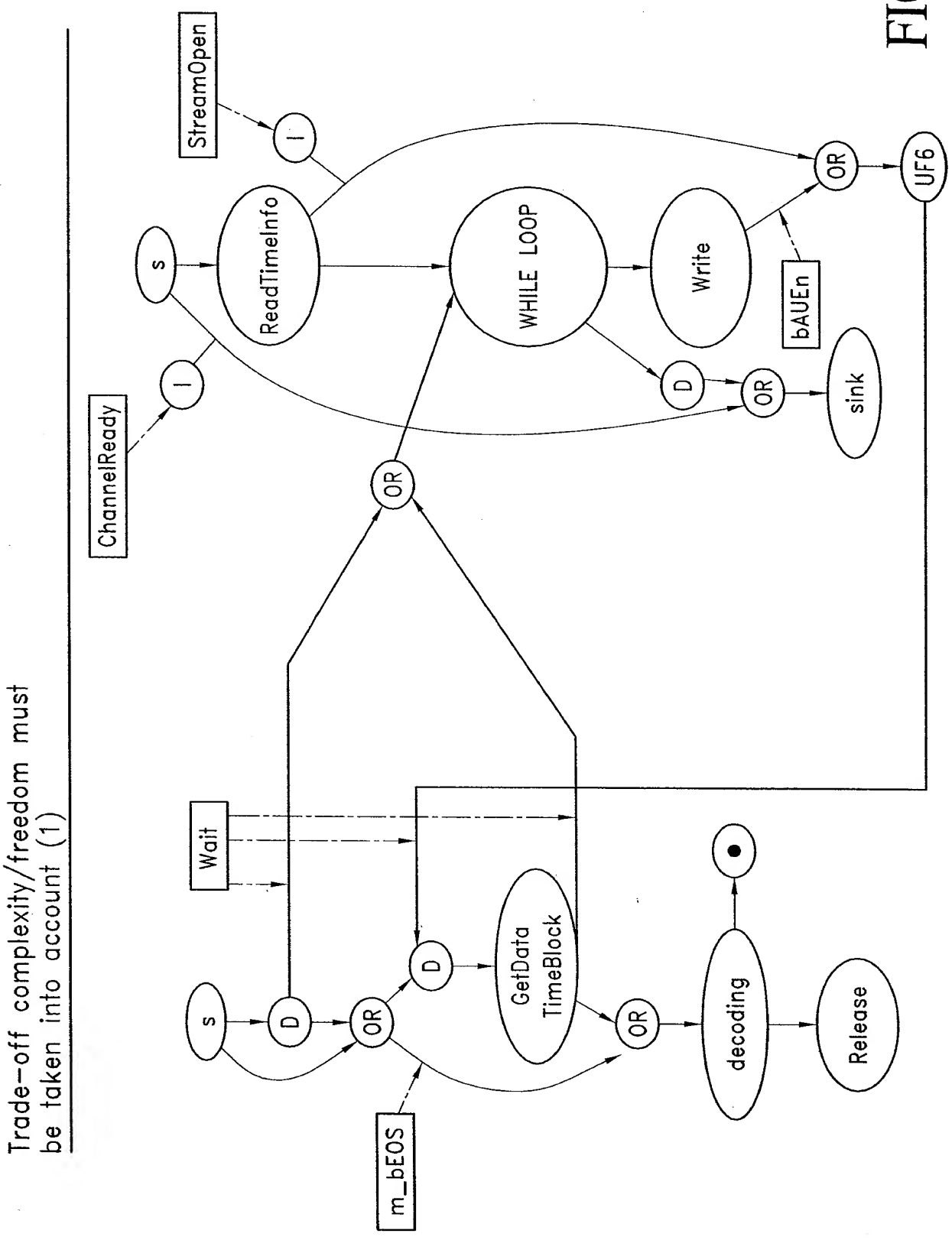

Trade-off complexity/freedom must be taken into account (1)
StreamOpen
ChannelReady
Wait
m_bEOS
s
I
D
OR
ReadTimeInfo
WHILE LOOP
Write
bAUEn
sink
UF6
GetData TimeBlock
decoding
Release

FIG. 22

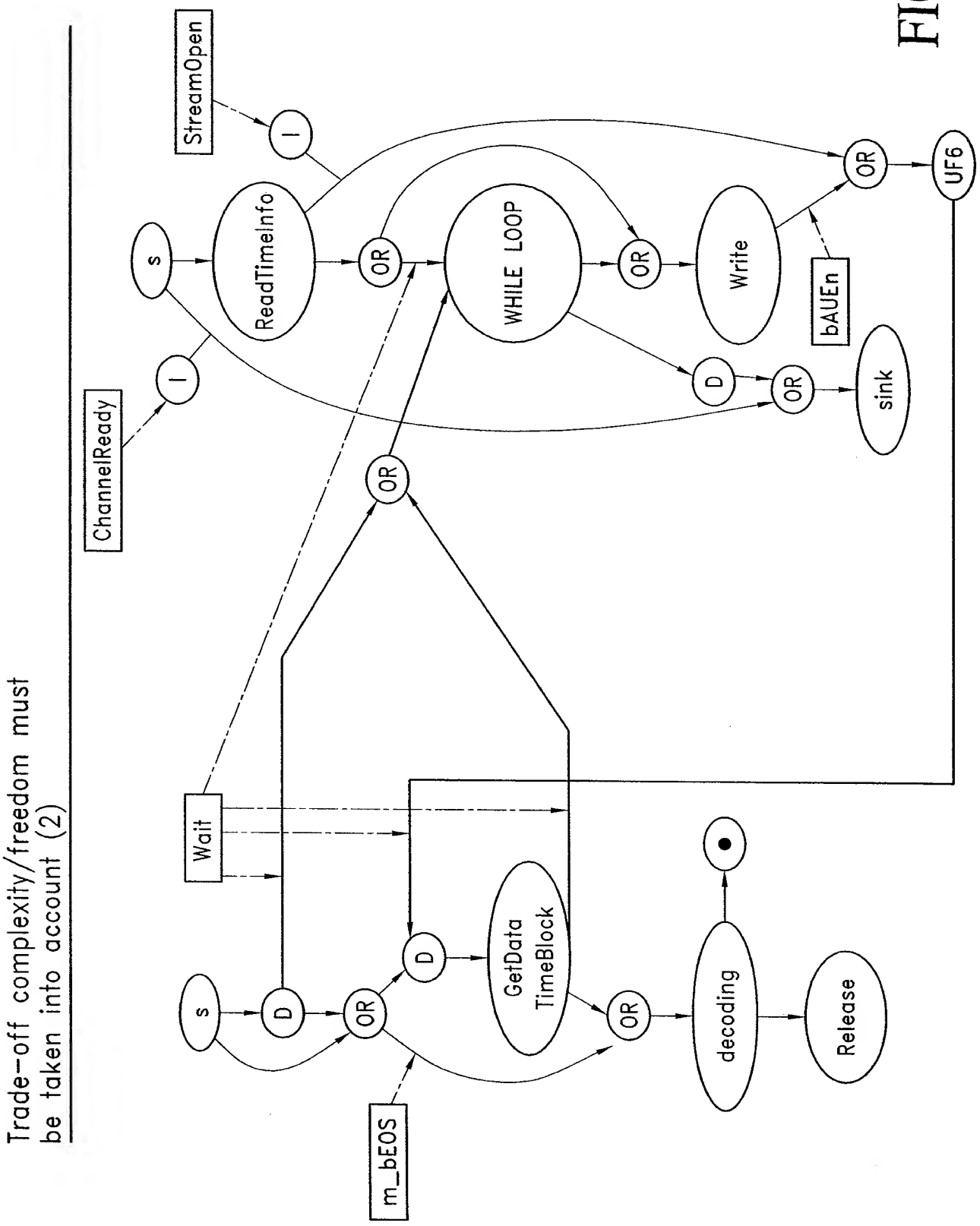

Trade-off complexity/freedom must be taken into account (2)
StreamOpen
ChannelReady
s
I
ReadTimeInfo
OR
WHILE LOOP
D
Write
bAUEn
sink
UF6
Wait
m_bEOS
GetData TimeBlock
decoding
Release

FIG. 23

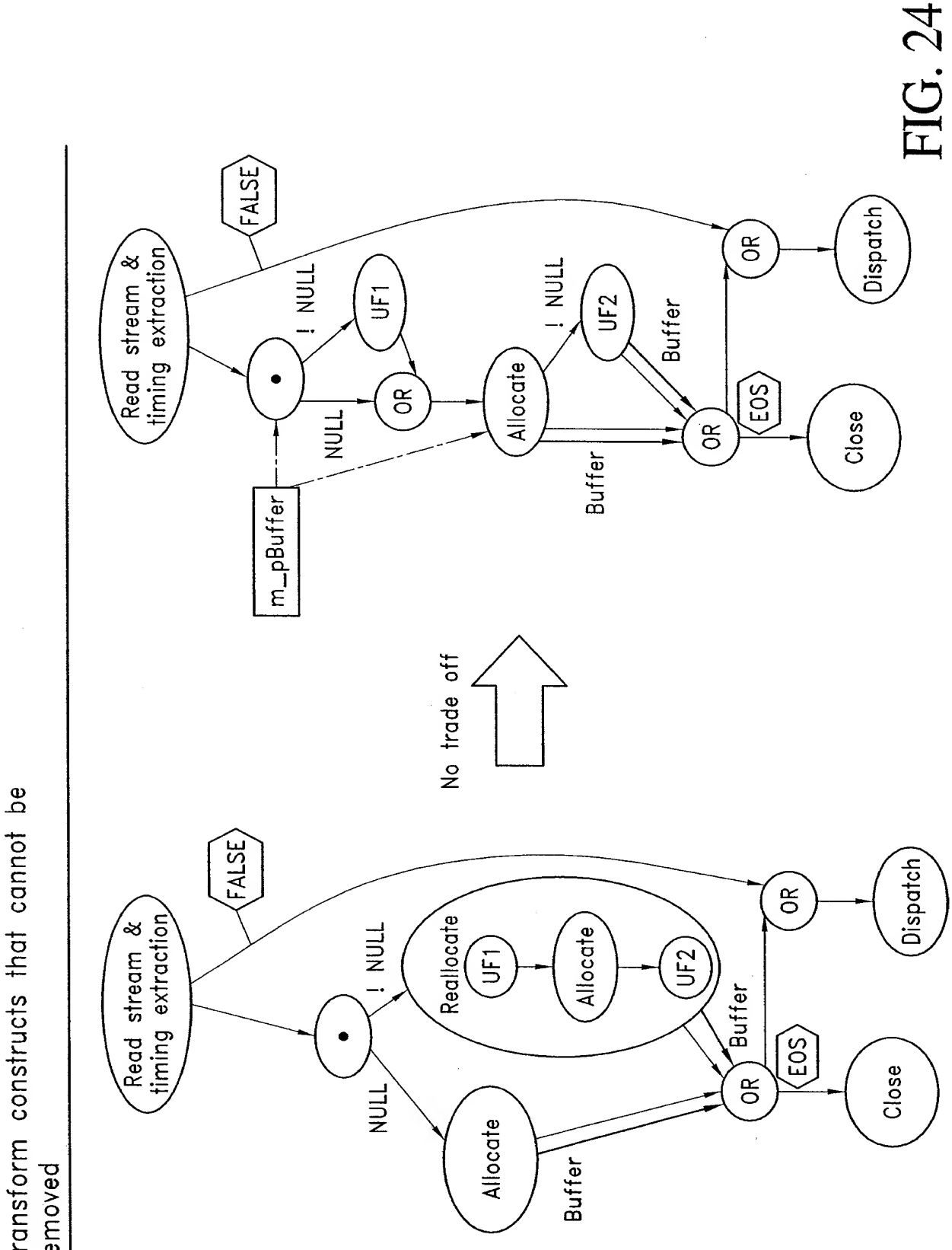

Transform constructs that cannot be removed
Read stream & timing extraction
FALSE
! NULL
NULL
Reallocate
UF1
Allocate
UF2
Buffer
OR
EOS
Close
Dispatch
No trade off
m_pBuffer

FIG. 24

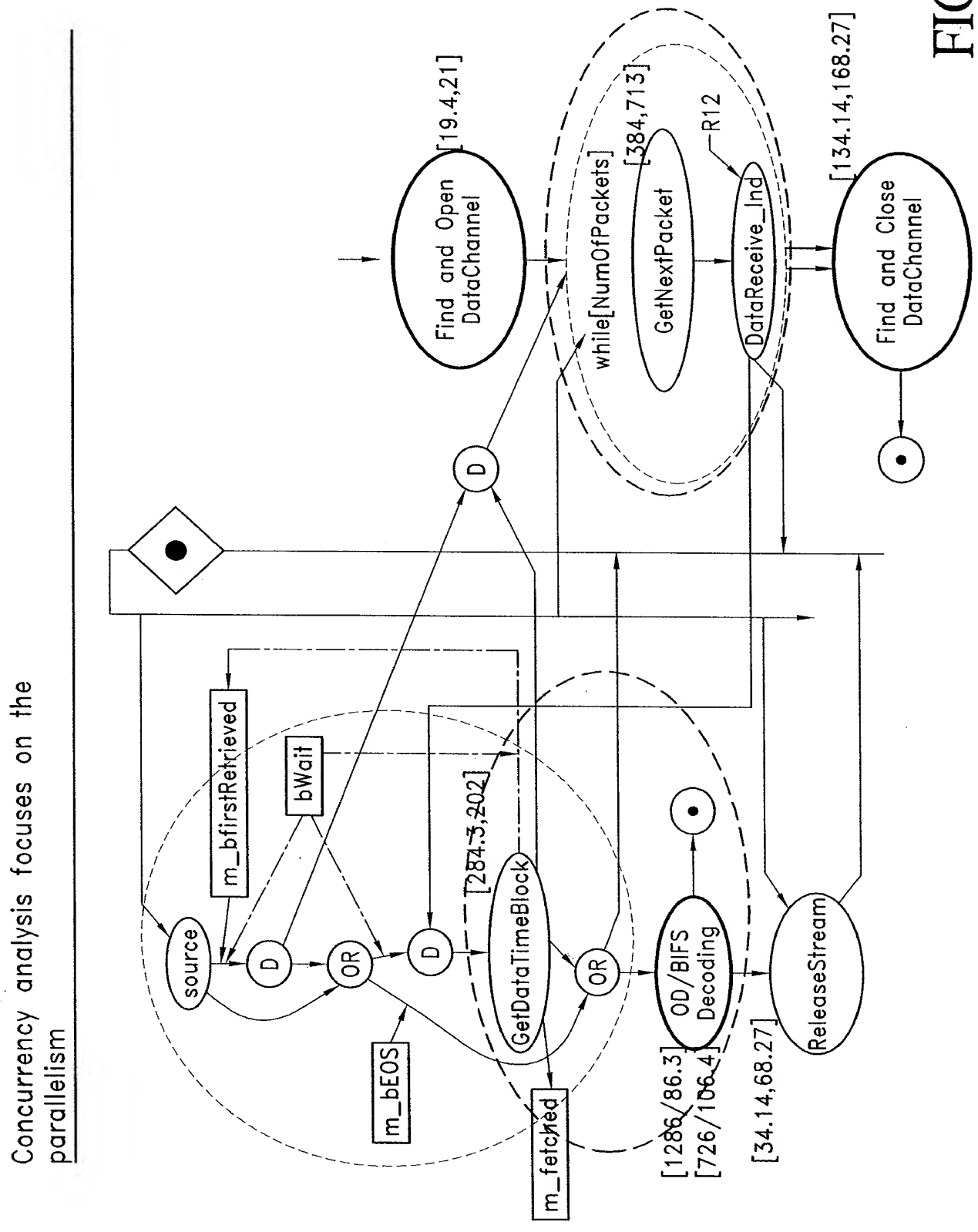

Concurrency analysis focuses on the parallelism
Find and Open DataChannel
[19.4,21]
while[NumOfPackets]
[384,713]
GetNextPacket
R12
DataReceive_Ind
[134.14,168.27]
Find and Close DataChannel
D
source
D
OR
D
[284.3,202]
GetDataTimeBlock
OR
OD/BIFS Decoding
ReleaseStream
m_bfirstRetrieved
bWait
m_bEOS
m_fetched
[1286/86.3]
[726/106.4]
[34.14,68.27]

FIG. 25

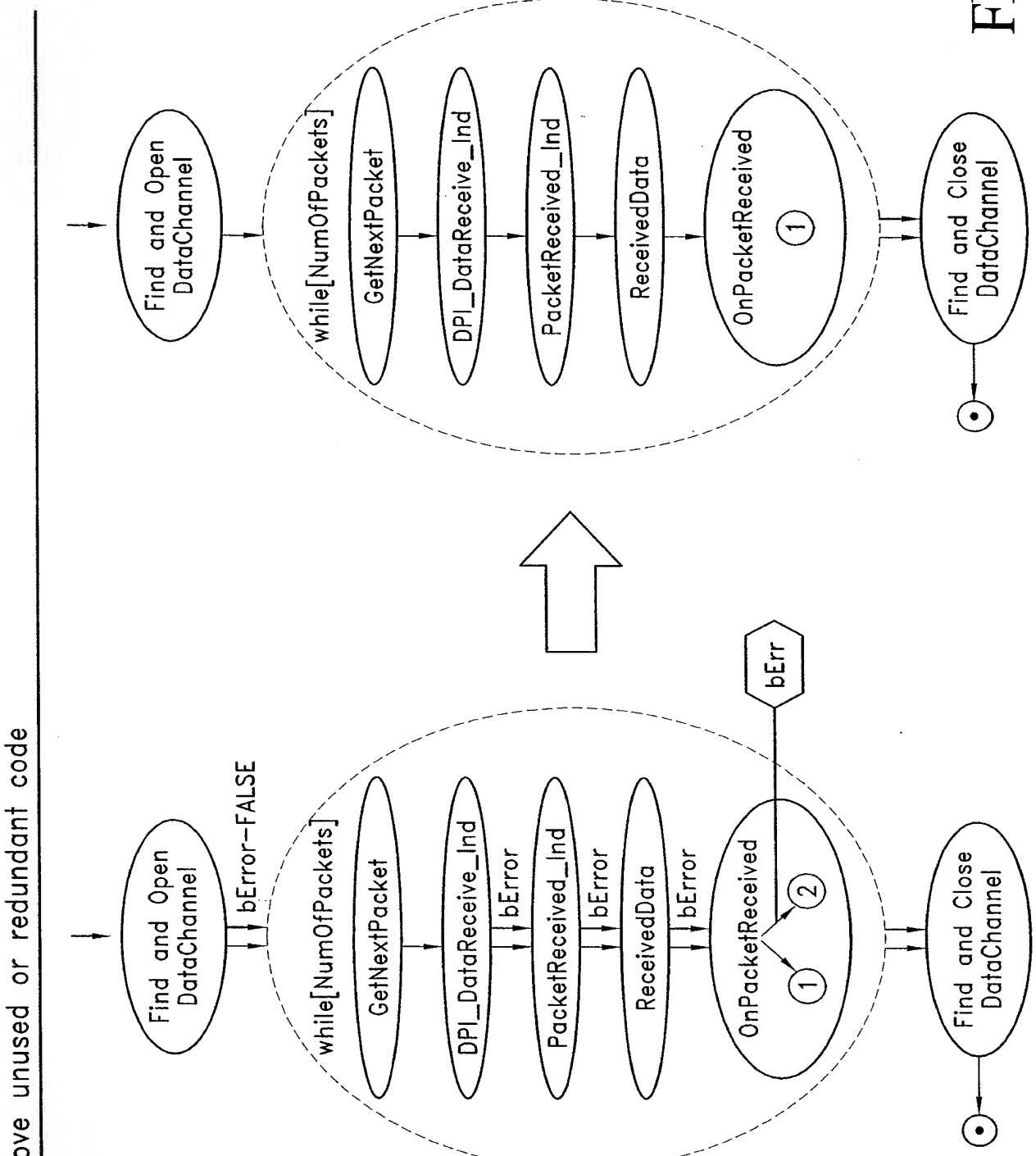

Remove unused or redundant code
Find and Open DataChannel
bError=FALSE
while[NumOfPackets]
GetNextPacket
DPI_DataReceive_Ind
bError
PacketReceived_Ind
bError
ReceivedData
bError
OnPacketReceived
1
2
bErr
Find and Close DataChannel
Find and Open DataChannel
while[NumOfPackets]
GetNextPacket
DPI_DataReceive_Ind
PacketReceived_Ind
ReceivedData
OnPacketReceived
1
Find and Close DataChannel

FIG. 26

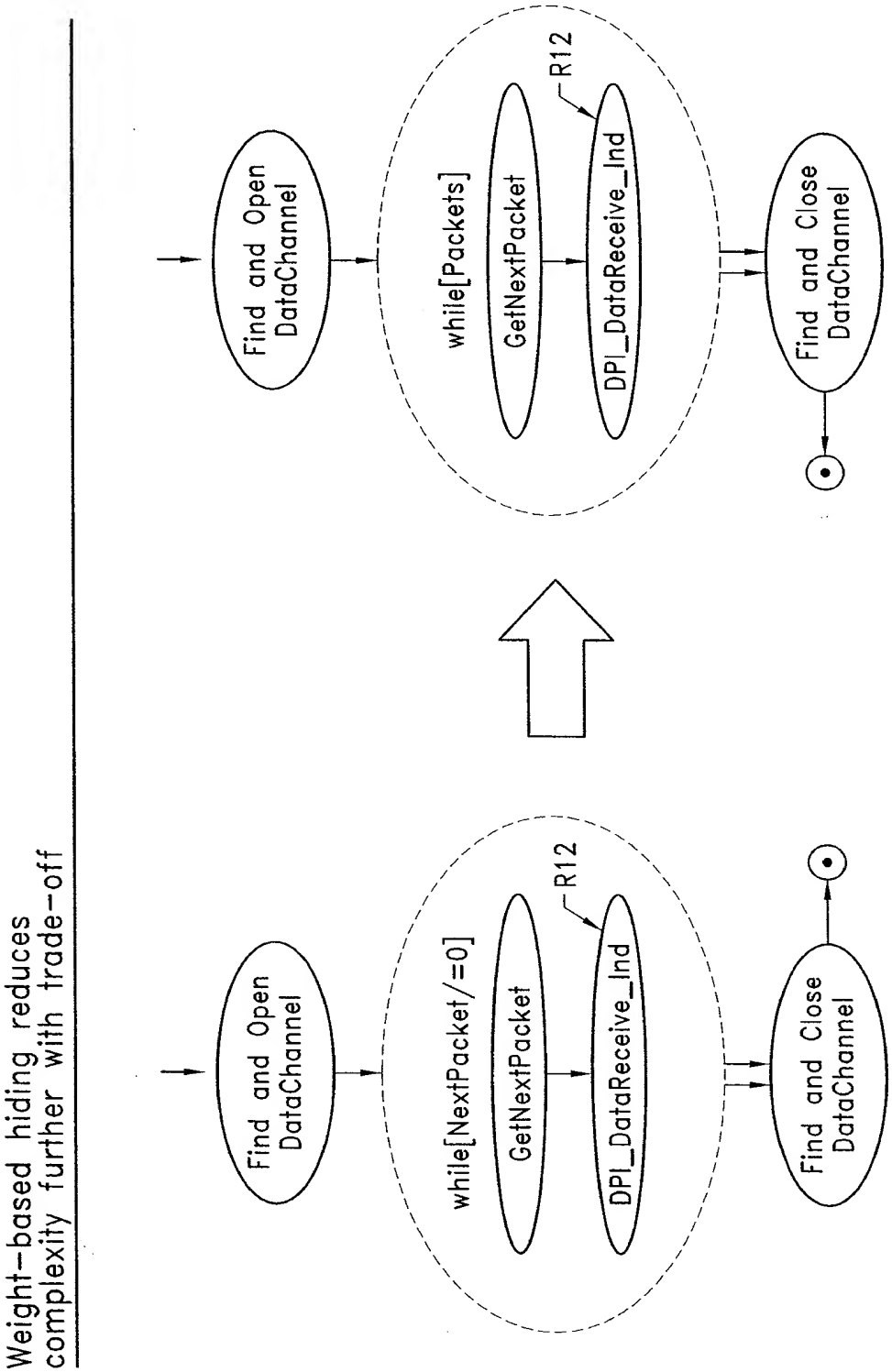

Weight-based hiding reduces complexity further with trade-off
Find and Open DataChannel
while[NextPacket/=0]
GetNextPacket
R12
DPI_DataReceive_Ind
Find and Close DataChannel
Find and Open DataChannel
while[Packets]
GetNextPacket
R12
DPI_DataReceive_Ind
Find and Close DataChannel

FIG. 27

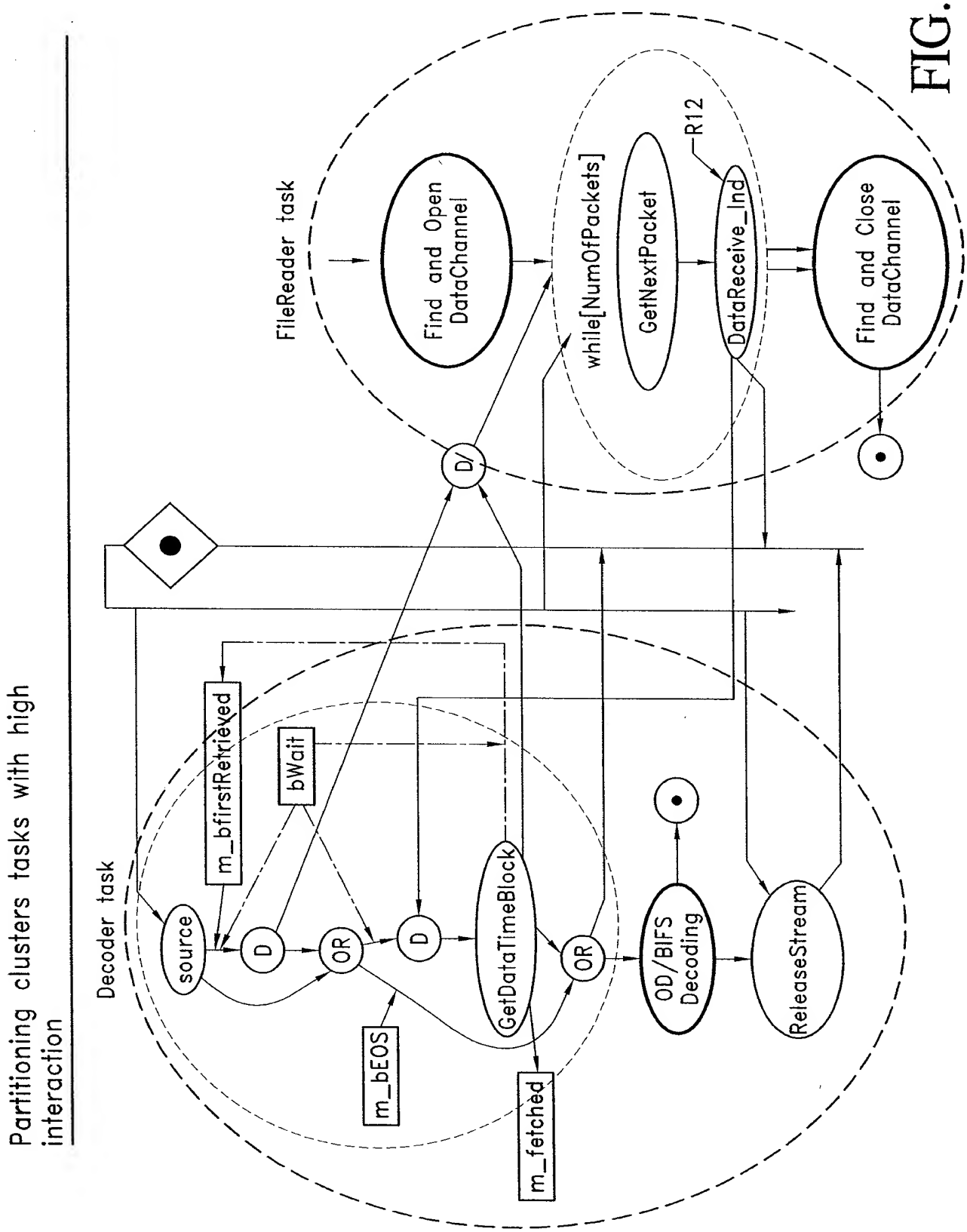

Partitioning clusters tasks with high interaction
FileReader task
Find and Open DataChannel
while[NumOfPackets]
GetNextPacket
R12
DataReceive_Ind
Find and Close DataChannel
D/
Decoder task
source
D
OR
D
GetDataTimeBlock
OR
OD/BIFS Decoding
ReleaseStream
m_bfirstRetrieved
bWait
m_bEOS
m_fetched

FIG. 28

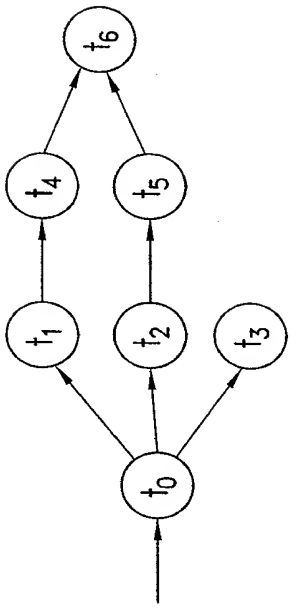

t_0
t_1
t_2
t_3
t_4
t_5
t_6

FIG. 29

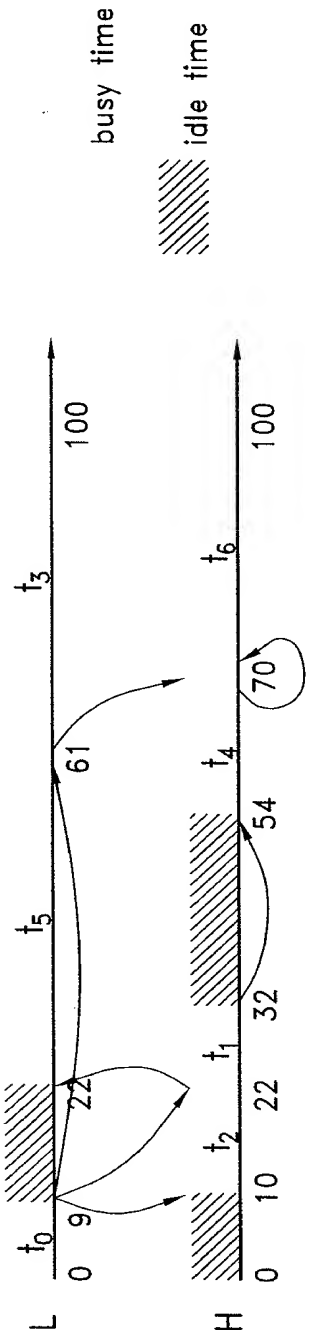

busy time
idle time
L
H
t_0
t_1
t_2
t_3
t_4
t_5
t_6
0
9
10
22
32
54
61
70
100
100

FIG. 30

FIG. 31

Thread Frame 1
Option 1
Thread Frame 2
Option 3
0
20
100
Thread Frame 1
Option 2
Thread Frame 2
Option 3
0
60
140

FIG. 32

Opt. 1
Opt. 1
Opt. 2
Opt. 2
Opt. n
Opt. n
Opt. 1
Opt. 2
Opt. n
$C_{S,1}$, $C_{T,1}$
$E_{S,1}$, $E_{T,1}$
$C_{S,2}$, $C_{T,2}$
$E_{S,2}$, $E_{T,2}$
$C_{S,n}$, $C_{T,n}$
$E_{S,n}$, $E_{T,n}$
1_1
1_2
1_n

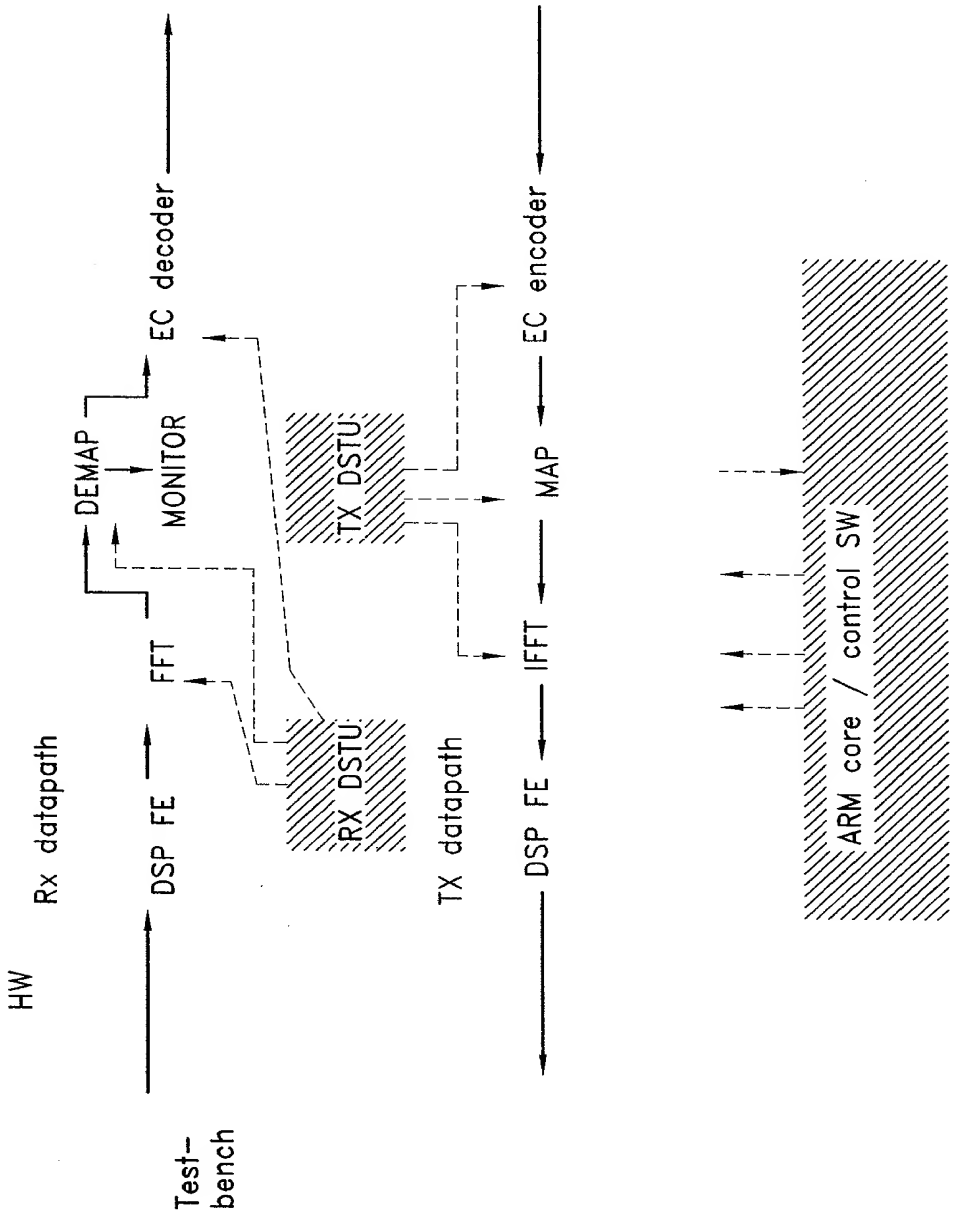

HW
Rx datapath
DSP FE
FFT
DEMAP
MONITOR
EC decoder
RX DSTU
TX DSTU
TX datapath
DSP FE
IFFT
MAP
EC encoder
ARM core / control SW
Test-bench

FIG. 33

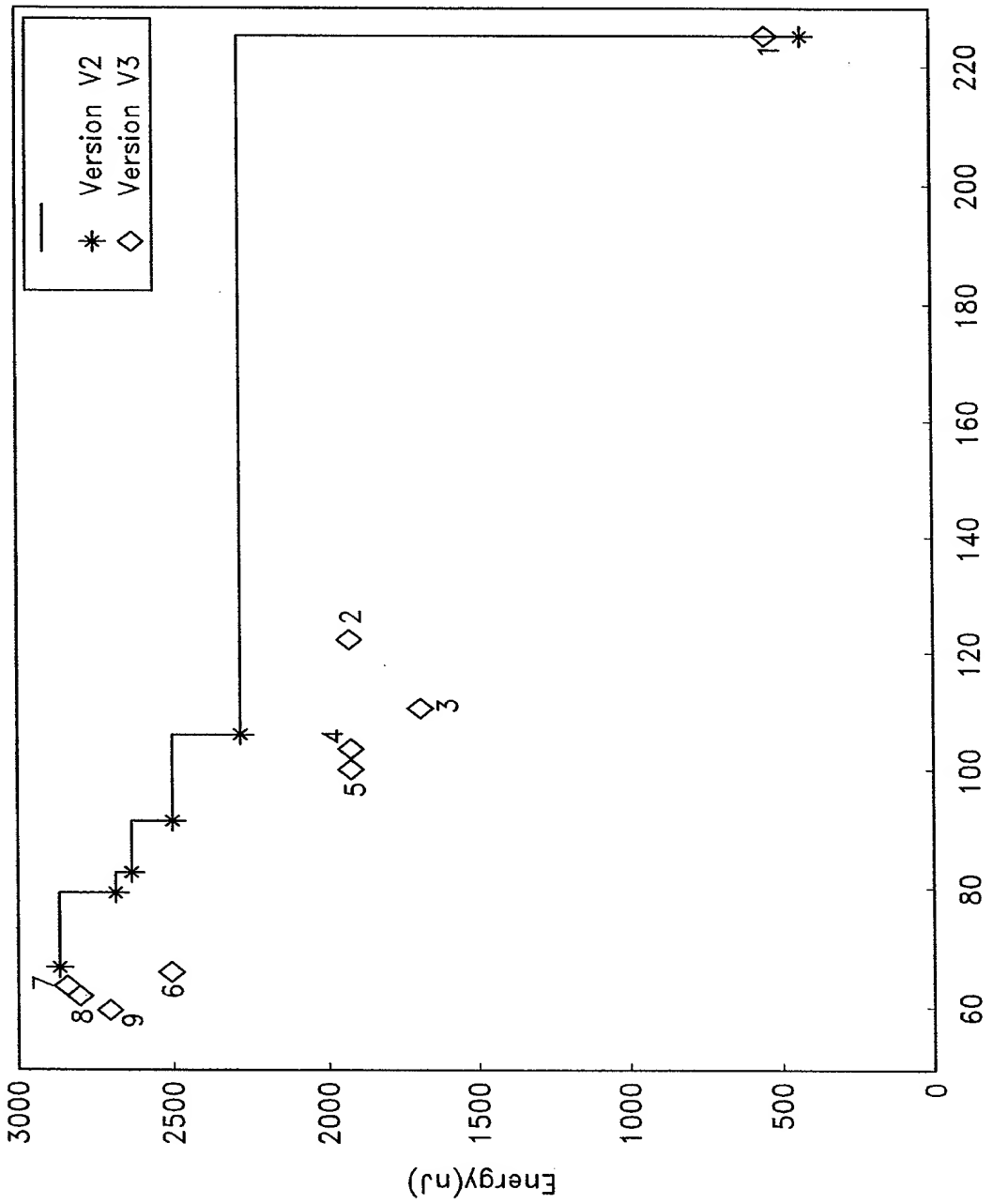

Version V2
Version V3
Energy(nJ)
Period(us)
3000
2500
2000
1500
1000
500
0
60
80
100
120
140
160
180
200
220
1
2
3
4
5
6
7
8
9

FIG. 34

FIG. 35

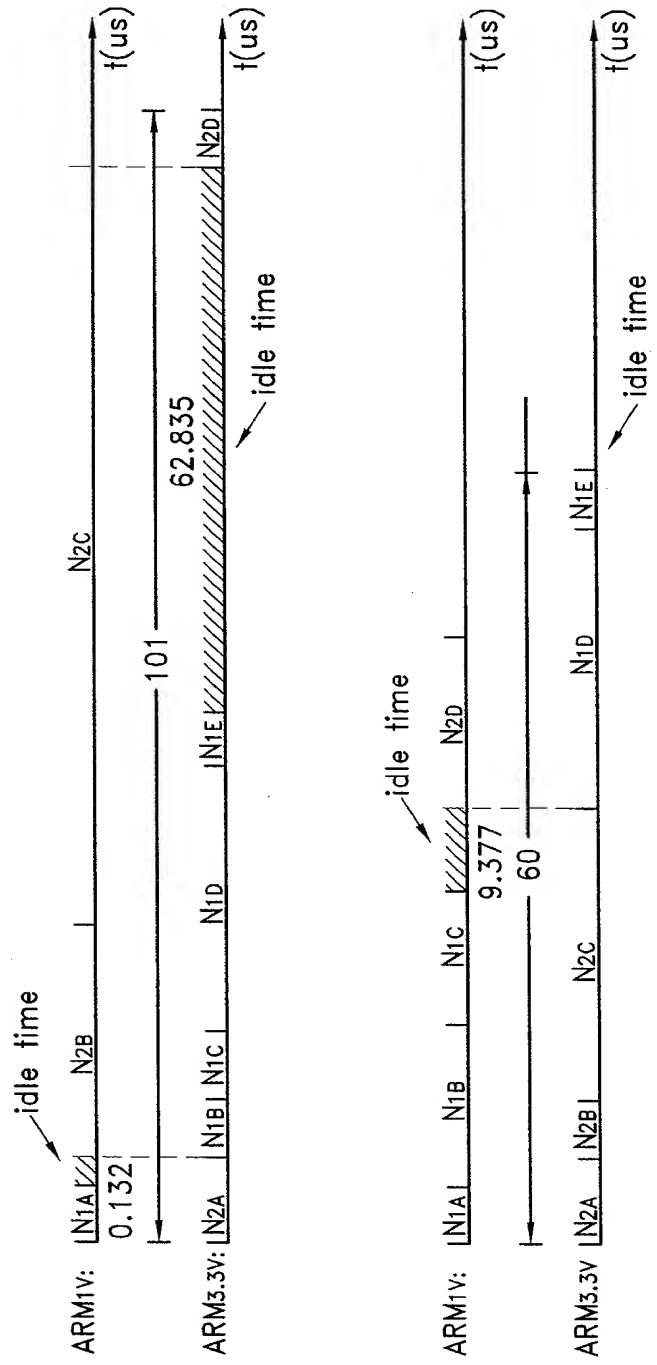
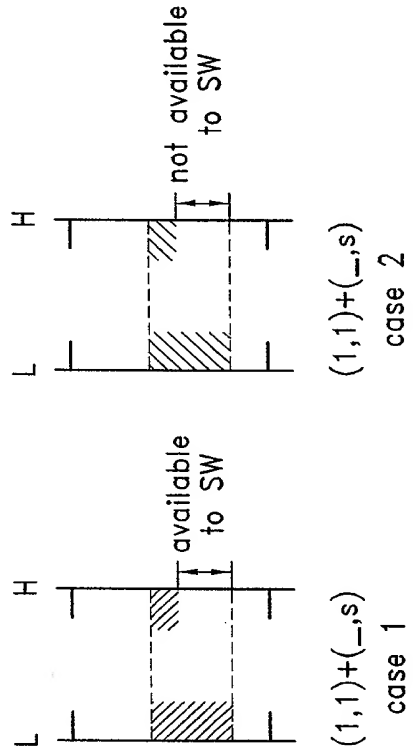

idle time
ARM1V:
N1A
N2B
N2C
t(us)
0.132
101
ARM3.3V:
N2A
N1B
N1C
N1D
N1E
62.835
N2D
idle time
t(us)
idle time
ARM1V:
N1A
N1B
N1C
N2D
t(us)
9.377
60
ARM3.3V
N2A
N2B
N2C
N1D
N1E
idle time
t(us)

FIG. 36


L
H
available
to SW
(1,1)+(_,s)
case 1
L
H
not available
to SW
(1,1)+(_,s)
case 2

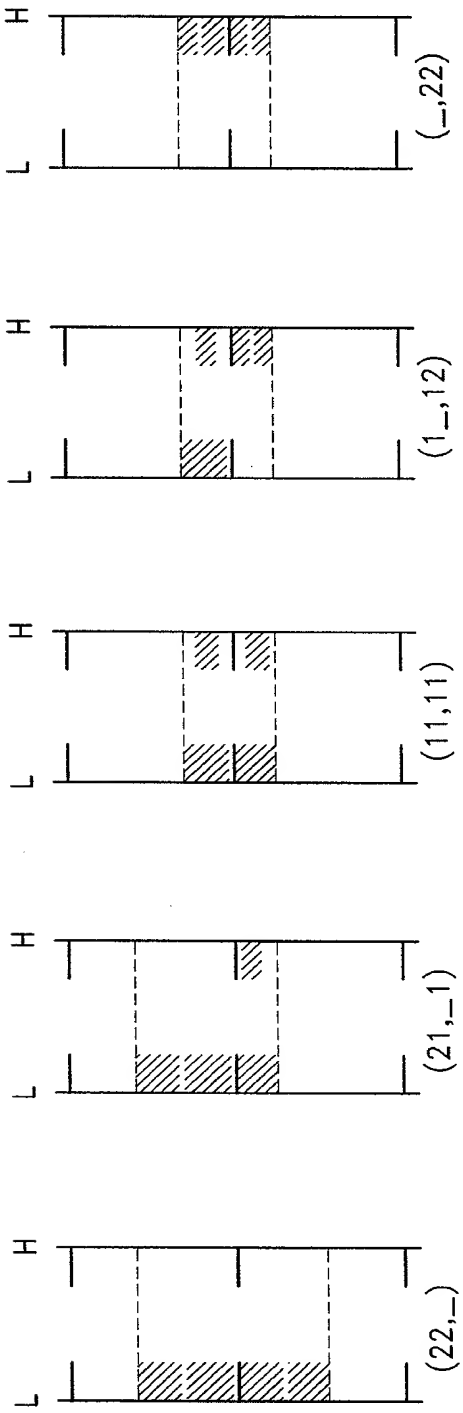


FIG. 37

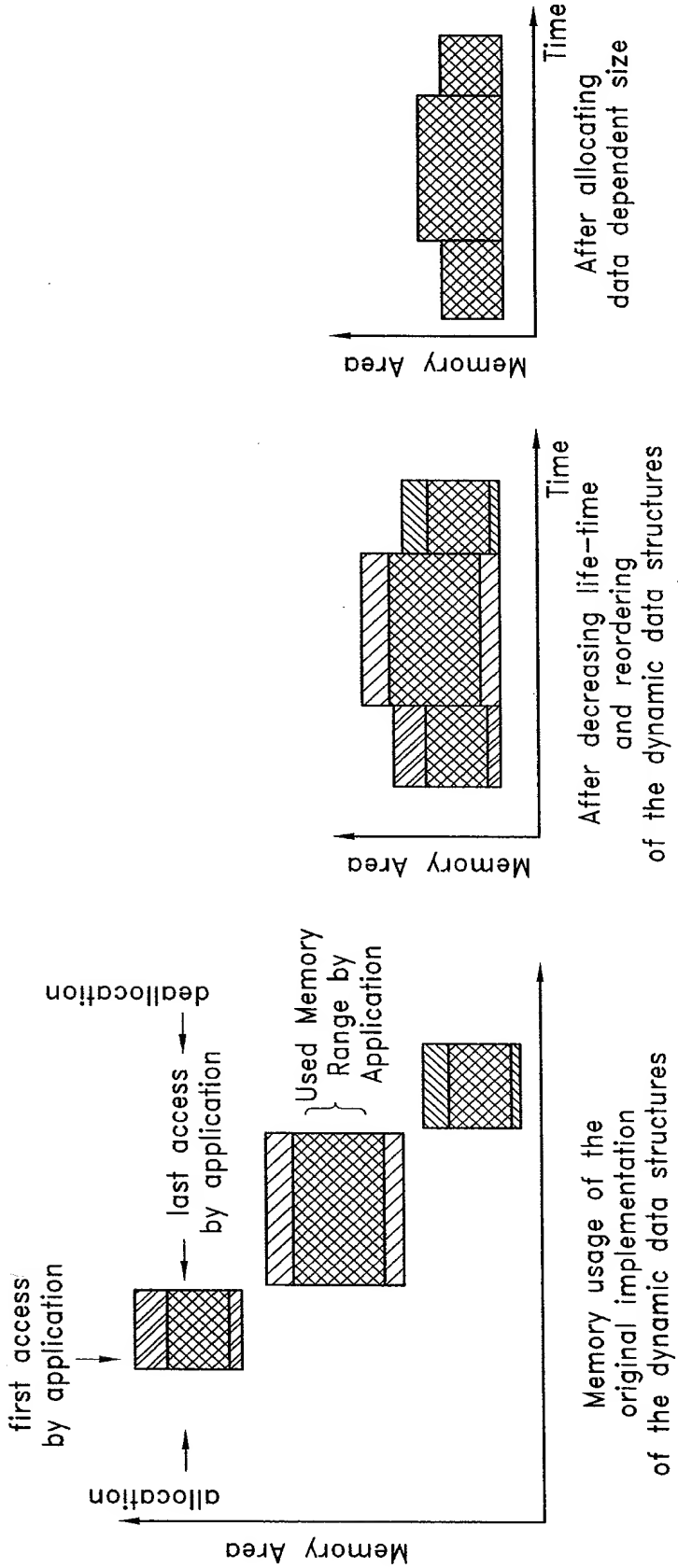

Memory Area
Time
After allocating
data dependent size
Memory Area
Time
After decreasing life-time
and reordering
of the dynamic data structures
Memory Area
allocation
first access
by application
last access
by application
deallocation
Used Memory
Range by
Application
Memory usage of the
original implementation
of the dynamic data structures

FIG. 38

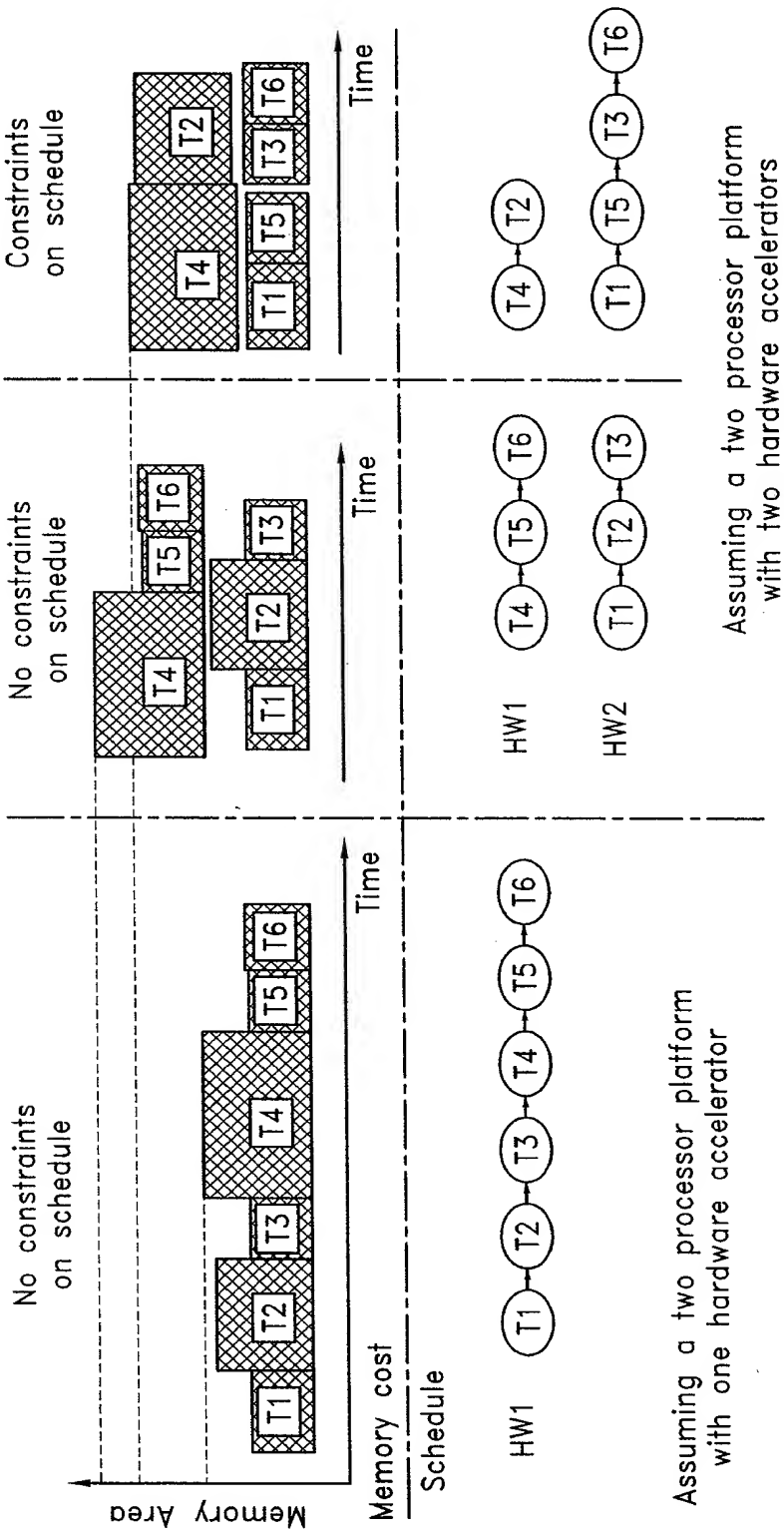

Memory Area
Time
Memory cost
Schedule
No constraints on schedule
T1
T2
T3
T4
T5
T6
HW1
Assuming a two processor platform with one hardware accelerator
No constraints on schedule
Time
HW1
HW2
Assuming a two processor platform with two hardware accelerators
Constraints on schedule
Time

FIG. 39

FIG. 40

	Execution Time							Energy Consumption						
	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_0	t_1	t_2	t_3	t_4	t_5	t_6
P1(H)	3	10	12	13	16	13	30	27	90	108	117	144	117	270
P2(L)	9	30	36	39	48	39	90	3	10	12	13	16	13	30

FIG. 41

	Thread Frame One			Thread Frame Two		
	opt.1	opt.2	opt.3	opt.1	opt.2	opt.3
Cycle Budget	20	60	100	40	60	80
Energy Cost	110	80	50	90	60	50

FIG. 42

Vdd	1 V	3 V	4 V	5 V
Frequency	10MHZ	30MHZ	40MHZ	50MHZ
Power (normalized)	1	27	64	125

FIG. 43

Task	4	5	6	7	8	9	10	11	12	14	15	16	17	18
Deadline (symbol)	128	96	128	128	768	256	128	132	16	64	16	2048	64	576
Ex. Time (ms)(10MHZ)	3	3	3	21	240	9	60	12	3	21	3	864	18	285

FIG. 44

(L,H)	1,1		2,_		_,2	
	s,_	_,s	s,_	_,s	s,_	_,s
Eq. $C_{S,i}$ (μs)	166	624	102	690	230	564
$E_{S,i}+E_{T,i}$ ($x10^{-6}$)	797	6247	230	6338	1364	6210

FIG. 45

(L,H)	1,1		2,_		_,2		Energy ($x10^{-6}$)
	s,_	_,s	s,_	_,s	s,_	_,s	
task4			30				6900
task7	1		65		62		100315
task8	1			138	629		$1.73x10^{6}$
task9			89				20470
task10			2	67	59		505582
task11			3	11	18		94960
task12			5		11		16154
task14			1	14	49		155798
task16	1		2	855	1190		$7.04x10^{6}$
task17			3	8	53		123686
task18	1		1	332	242		$2.44x10^{6}$
						Total	$1.22x10^{7}$

FIG. 46

(L,H)	1,1		2,_		_,2	
	s,_	_,s	s,_	_,s	s,_	_,s
Eq. $C_{S,i}$ (μs)	166	498	102	306	188	564
$E_{S,i}+E_{T,i}$ ($x10^{-6}$)	797	5113	230	2882	1322	6210

FIG. 47

(L,H)	1,1		2,_		_,2		Energy ($x10^{-6}$)
	s,_	_,s	s,_	_,s	s,_	_,s	
task4			30				6900
task7	123		4		1		100273
task8	429	339					$2.075x10^6$
task9			89				20470
task10	10	117	1				606421
task11	12	19				1	112921
task12	14	1			1		17593
task14	32	30			1	1	186426
task16	467	1579		2			$8.45x10^6$
task17	40	22	1	1			147478
task18	4	570		1	1		$2.92x10^6$
						Total	$1.464x10^7$

FIG. 48

(L,H)	22,___		21,_1		11,11		1_,12		__,22	
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s
Eq. $C_{S,i}$ (μs)	204	612	268	804	332	996	352	1056	376	1128
$E_{S,i}+E_{T,i}$ ($x10^{-6}$)	460	5764	1027	7975	1594	10226	2117	11269	2644	12420

FIG. 49

(L,H)	22,__		21,_1		11,11		1_,12		__,22		Energy ($x10^{-6}$)
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	
task4	15										6900
task7			4		69		1				100271
task8				2	213	168	1				$2.076x10^6$
task9	45										20700
task10				2	5	57					606802
task11					6	9		1			112867
task12			2	1	5						17999
task14			1	1	14	15	1				186825
task16					234	787	1	2			$8.446x10^6$
task17					20	10	1	1			147526
task18					2	284	1			1	$2.92x10^6$
										Total	$1.464x10^7$

FIG. 50

(L,H)	222,____		221,___1		121,1_1		12_,1_2		_2_,2_2	
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s
Eq. $C_{S,i}$ (μs)	204	816	268	1072	332	1328	364	1456	396	1584
$E_{S,i}+E_{T,i}$ ($x10^{-6}$)	588	13440	1612	18496	2636	23552	3628	26560	4620	29568

FIG. 51

(L,H)	222,___		221,__1		121,1_1		12_,1_2		_2_,2_2		Energy ($x10^{-6}$)
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	
task4	15										8820
task7					33	7	1		1		260100
task8		1			99	154	1	1			$3.93x10^6$
task9	45										26460
task10						25		1		16	$1.09x10^6$
task11				1	1	8					209548
task12	1	1			2	1					42852
task14					5	14	1		1		351156
task16					41	639	1	1			$1.52x10^7$
task17					8	11	1		1		288408
task18						74		1		117	$5.23x10^6$
										Total	$2.665x10^7$

FIG. 52

(L,H)	2222,____		2221,___1		1221,1__1		122_,1__2		_22_,2__2	
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s
Eq. $C_{S,i}$ (μs)	204	1020	268	1340	332	1660	370	1850	408	2040
$E_{S,i}+E_{T,i}$ ($x10^{-6}$)	716	26012	2065	35297	3710	44878	5309	51189	6908	57500

FIG. 53

(L,H)	2222,____		2221,___1		1221,1__1		122_,1__2		_22_,2__2		Energy ($x10^{-6}$)
	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	s,_	_,s	
task4	15										10740
task7				1	33	7			1		441681
task8			1	1	57	132	1				$6.18x10^6$
task9	45										32220
task10				1				24		7	$1.67x10^6$
task11				4		4					320700
task12			1		1		1			1	68584
task14				1	3	11			1		546993
task16				1		437		72		2	$2.34x10^7$
task17				6	5	5					454722
task18								46		98	$7.99x10^6$
										Total	$4.11x10^7$

FIG. 54

Case Number	1	2	3	4	5
Total Energy Cost	12.2	14.64	14.64	26.65	41.1

FIG. 55

Case Number	1	2	3	4	5
Total Energy Cost	31.2	34.6	34.6	39.1	41.1

FIG. 56

	1HW accelerator		2HW accelerators	
	Execution Time	Processor Energy	Execution Time	Processor Energy
OD	9ms	0.37mJ	9ms	0.37mJ
BIFS	24ms	0.98mJ	24ms	0.98mJ
Delivery	8.1ms	0.32mJ	16.2ms	0.64mJ
Wavelet	30ms	1mJ	30ms	2mJ
Total	30ms	2.58mJ	30ms	4mJ

FIG. 57

	1HW accelerator			2HW accelerators		
	Mem. Accesses	Mem. Size Pre	Mem. Size Post	Mem. Accesses	Mem. Size Pre	Mem. Sixe Post
OD	0.58k	10kB	0.58kB	0.58kB	10kB	0.58kB
BIFS	2.41k	41kB	2.41kB	2.41k	41kB	2.41kB
Delivery	35.9k	35.9kB	12.4kB	71.8k	71.8kB	17kB
Wavelet	35.9k	35.9kB	12.4kB	71.8k	71.8kB	17kB
Total	74.9k	86.9kB	14.8k	146k	193kB	19.4k

1HW accelerator		2HW accelerators	
Energy Pre	Energy Post	Energy Pre	Energy Post
0.78mJ	0.16mJ	1.54mJ	0.19mJ